NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

MANDIBULAR ADVANCING POSITIVE PRESSURE APNEA REMEDIATION DEVICE (MAPPARD)

by

Benjamin T. Morehead

June 2014

Thesis Advisor: Joseph A. Sullivan
Second Reader: Jeffrey D. Weekley

This thesis was performed at the MOVES Institute

Approved for public release; distribution is unlimited

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	**2. REPORT DATE** June 2014	**3. REPORT TYPE AND DATES COVERED** Master's Thesis

4. TITLE AND SUBTITLE
MANDIBULAR ADVANCING POSITIVE PRESSURE APNEA REMEDIATION DEVICE (MAPPARD)

5. FUNDING NUMBERS

6. AUTHOR(S) Benjamin T. Morehead

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Naval Postgraduate School
Monterey, CA 93943-5000

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES)
N/A

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government. IRB protocol number ____N/A____.

12a. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited

12b. DISTRIBUTION CODE
A

13. ABSTRACT (maximum 200 words)

Current research has shown that an increasing number of returning troops from deployments are being diagnosed with obstructed sleep apnea (OSA). OSA causes excessive daytime sleepiness that can endanger the readiness of Soldiers by impacting concentration, decision-making skills, personality change, hypertension, depression, headaches and has been shown to contribute to cardiovascular disease.

A main factor for remediation of OSA is Soldier's compliance with prescribed treatment plans. The two most popular methods for OSA treatment are continuous positive airway pressure (CPAP) and the Mandibular Advancing Device (MAD). Both of these devices have known compliance issues, which keep treatment of OSA to roughly 60 percent of those prescribed.

We utilized a parameterized upper airway fluid structure–interaction (FSI) simulation to validate our hybrid OSA device (MAPPARD), which addressed the compliance issues found in typical OSA treatment devices. While being 25 percent less advanced than the MAD device and 50 percent less pressure than the CPAP device, our MAPPARD performed better than either current device, thus showing potential to improve Soldier treatment compliance. This study contributes to the ongoing exploration of the role of modeling and simulations for testing and evaluation of medical devices.

14. SUBJECT TERMS Obstructed sleep apnea, remediation devices, modeling and simulations, health care simulations, mandibular advancing positive pressure apnea remediation device(MAPPARD), mandibular advancing device(MAD), obstructed sleep apnea (OSA)	**15. NUMBER OF PAGES** 115
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	**18. SECURITY CLASSIFICATION OF THIS PAGE**	**19. SECURITY CLASSIFICATION OF ABSTRACT**	**20. LIMITATION OF ABSTRACT**
Unclassified	Unclassified	Unclassified	UU

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

MANDIBULAR ADVANCING POSITIVE PRESSURE APNEA REMEDIATION DEVICE (MAPPARD)

Benjamin T. Morehead
Major, United States Army
B.S., University of Central Florida, 1996

Submitted in partial fulfillment of the
requirements for the degree of

**MASTER OF SCIENCE IN
MODELING, VIRTUAL ENVIRONMENTS AND SIMULATION**

from the

**NAVAL POSTGRADUATE SCHOOL
June 2014**

Author: Benjamin T. Morehead

Approved by: Joseph A. Sullivan
Thesis Advisor

Jeffrey D. Weekley
Second Reader

Peter J. Denning
Chair, Department of Computer Science

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

Current research has shown that an increasing number of returning troops from deployments are being diagnosed with obstructed sleep apnea (OSA). OSA causes excessive daytime sleepiness that can endanger the readiness of Soldiers by impacting concentration, decision-making skills, personality change, hypertension, depression, headaches and has been shown to contribute to cardiovascular disease.

A main factor for remediation of OSA is Soldier's compliance with prescribed treatment plans. The two most popular methods for OSA treatment are continuous positive airway pressure (CPAP) and the Mandibular Advancing Device (MAD). Both of these devices have known compliance issues, which keep treatment of OSA to roughly 60 percent of those prescribed.

We utilized a parameterized upper airway fluid structure–interaction (FSI) simulation to validate our hybrid OSA device (MAPPARD), which addressed the compliance issues found in typical OSA treatment devices. While being 25 percent less advanced than the MAD device and 50 percent less pressure than the CPAP device, our MAPPARD performed better than either current device, thus showing potential to improve Soldier treatment compliance. This study contributes to the ongoing exploration of the role of modeling and simulations for testing and evaluation of medical devices.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF ACRONYMS AND ABBREVIATIONS

API	Application Programming Interface
CAD	computer-aided design
CPAP	continuous positive airway pressure
CSA	cross-sectional area
CNC	computer numerically controlled
DES	discrete-event simulation
DOD	Department of Defense
FEM	finite element mesh
FSI	fluid-structure interaction
HPRC	Human Performance Resource Center
KISSlicer	Keep It Simple Slicer
LAUP	laser-assisted uvulopalatopharyngoplasty
MAD	Mandibular Advancing Device
MAPPARD	Mandibular Advancing Positive Pressure Apnea Remediation Device
MMA	maxilla-mandibular-advancement
MOVES	Modeling, Virtual Environments, and Simulation
NPS	Naval Postgraduate School
NURBS	non-uniform rational basis spline
OSA	obstructive sleep apnea
OPAL	oral, pharyngeal and laryngeal
OP	oropharynx
REM	rapid eye movement
UBC	University of British Columbia
UPPP	uvulopalatopharyngoplasty
VA	Veterans Association
VP	velopharynx

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

I would like to thank CDR Joe Sullivan for giving me continued support and advice during the development of this thesis. His interest in this topic created some very interesting discussions, which ultimately prompted me to do more than I thought was possible. Without his help, this thesis would not be possible.

I would also like to thank Jeffrey Weekley for his help and guidance not only on this thesis, but also in the vast world of computer modeling. Without his help, there would not be a quality model for displaying our MAPPARD device within the thesis.

I would like to thank my brother, Dr. Brady Morehead, for his crazy idea for this device. His curiosity led to us down the path of this research, and allowed me to use my modeling and simulation knowledge on something that interested us both.

I need to thank Peter Anderson, whose FSI simulation allowed me to test and validate my device. Many, many hours of his time were spent in video chat with me going over the intricacies of his code and simulation, all while he was trying to complete his dissertation. His selflessness did not go unnoticed.

Special thanks to Ricardo Bastos for assisting me in developing the 3D model in Blender, and Jeff Houde for volunteering his time and resources to 3D print the prototype device. Their support was essential to having quality products to display within this thesis.

Lastly, I would be remiss if I did not thank my wife, Rachale, and my kids, Aaron and Hannah, for dealing with me these last two years. I have been stressed out and inconsiderate to their needs, yet they have shown nothing but support and tolerance. Their love has allowed me to go forever onward and upward.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. RESEARCH MOTIVATION

Research has found that there is a recent spike in the number of veterans (upwards of 61 percent) receiving disability benefits from being diagnosed with obstructive sleep apnea (Brook, 2010). Obstructive sleep apnea is the leading sleep disorder today (Thorpy & Yager, 2001) and causes many quality of life and systemic medical issues. As a Modeling, Virtual Environments, and Simulation (MOVES) graduate student and U.S. Army officer who is diagnosed with obstructed sleep apnea, this study has motivated me to explore solutions within modeling and simulations that can improve Soldier readiness and compliance issues.

Additionally, this research will help validate the role of modeling and simulations in the test and evaluation of medical devices. The use of modeling and simulations can potentially save considerable monetary resources in development costs due to reduced “live” human testing requirements. We hope our research will accelerate the adoption of simulations within the healthcare community.

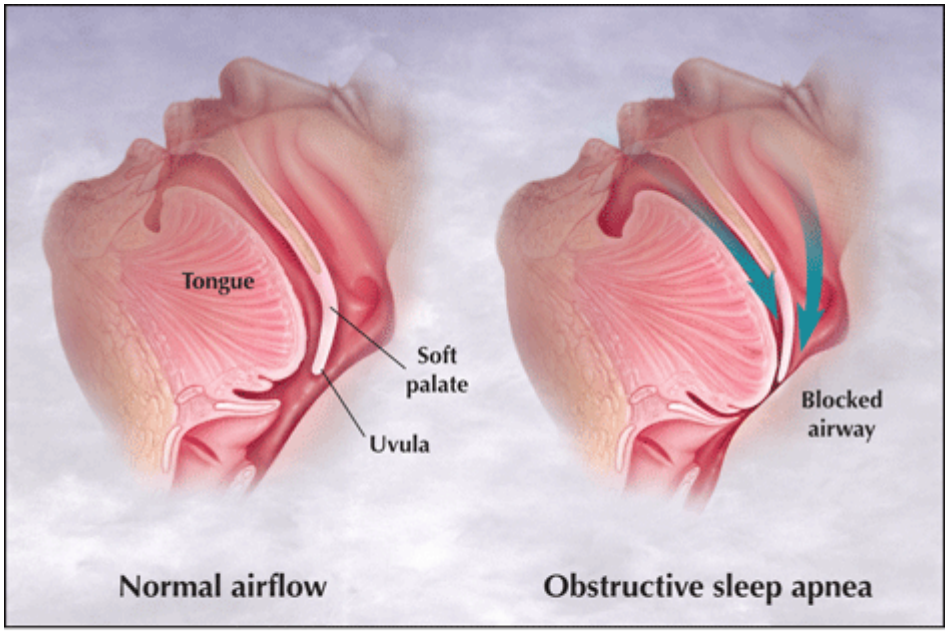
THIS PAGE INTENTIONALLY LEFT BLANK

II. BACKGROUND

A. OBSTRUCTIVE SLEEP APNEA

Obstructive sleep apnea (OSA) is a disease characterized by reoccurring episodes of partial or complete blockage of the upper airways or pharyngeal airspace during sleep (Ito, Cheng, Shih, Koomullil, Soni, & Waite, 2011). The blockage causes a cessation in the breathing rhythm, which is an apnea event. The three types of apnea events are 1) “obstructive” event—when there is a compromised or completely closed upper airway; 2) “central” event- reduction or cessation of brain stem respiratory motor output; and 3) a combination of both central and obstructive events (Dempsey, Veasey, Morgan, & Donnell, 2010). The obstructive apnea is the most common, and will be the type of apnea event that we will concentrate on during this study.

The diagnosis of OSA is usually performed with a sleep study, or polysomnogram, which will reveal the amount of apneas, their duration, and sleep stage of occurrence. Most of the apnea episodes occur within the supine or rapid eye movement (REM) sleep stage (Thorpy & Yager, 2001). The apnea is caused when the tongue and throat muscles relax and press the uvula and throat shut, thus blocking the airway (see Figure 1). The blockage creates an inability of the air to move in or out of the lungs, which results in decreased oxygen levels within the blood. This cessation, or lack, of breathing can last only a few seconds or up to a couple minutes. It is only interrupted when the person will partially awaken and resume breathing. When individuals do not remain in the REM sleep stage, they miss out on important regenerative functions of normal sleep. As a person loses more sleep or continues to receive less than optimal amounts of sleep, daytime sleepiness and sleep deprivation symptoms increase. Most people with OSA do not fully awaken during the episode and usually do not know they have the ailment until tested or a bed partner explains unusual breathing observed. It is estimated that 80–90 percent of those affected with OSA suffer undiagnosed (Haskell, McCrillis, Haskell, Scarfe, & Farman, 2009).



Obstructive sleep apnea occurs when muscles at the back of the throat relax and obstruct airflow. These muscles normally support your tonsils and tongue, the soft palate on the roof of your mouth, and the uvula, which hangs from the soft palate at the back of your mouth.

Figure 1. Obstructive sleep apnea (from "Obstructive Sleep Apnea," n.d.).

Surprisingly, patients with OSA suffer little to no problems with their breathing or airway while they are awake (Dempsey et al., 2010), and their healthy control systems are sufficiently sensitive enough to coordinate chest wall and upper airway to provide proper airway diameter for respiratory needs. When patients do fall asleep, they lose the neuronal activation of the dilator muscles, which cause the collapse of the pharynx. The result is a repeated cyclic OSA event through multiple compensatory processes, as the body awakens to overcome the apnea- only to fall asleep and lose the neuronal activation once again.

The population cross-sectional parameters that are the most common identifiable risk factors with OSA are excess body weight (the dominant contributor), male gender, aging, and cranial facial structures (Dempsey et al., 2010). These factors are found to have a high correlation between OSA and

patient, but are not a conclusive reason or diagnoses tool for evaluating a patient until a polysomnogram is conducted.

Most OSA patients complain about being tired and fatigued during the day. While this may pose a dangerous side effect of OSA, experts explain many behavioral and physical consequences that can be attributed to a considerable amount of other effects. Known behavioral effects of OSA are daytime sleepiness, impaired concentration, memory (Ito et al., 2011), creativity, moral judgment, and decision-making (Tucker, Whitney, Belenky, Hinson, & Van Dongen, 2010). Perhaps more important are the physical effects of OSA, which include cardiovascular disorders, such as myocardial infarction and hypertension, decreased immune response, and cerebrovascular disease (Ballard, Gay, & Strollo, 2007; Johnson, Broughton, & Halberstadt, 2003). Other common ailments of OSA patients are:

1. High blood pressure—due to increased cardiovascular effort during apnea episodes. Around 30 percent of patients being treated for high blood pressure have OSA syndrome (Johnson et al., 2003).
2. Joint and muscle pains—lack of regenerative sleep can cause aches and pains that are common with OSA and other sleep disorders.
3. Overweight—Weight gain can cause narrowing of the upper airway, and can cause obstruction by the collapse of the tongue and neck (Thorpy & Yager, 2001). Studies have shown that 70 percent of patients that have OSA are 20 percent overweight, thus not being a determining factor but significant nonetheless (Reite, Ruddy & Nagel, 2002).
4. Snoring—Snoring is a sign of upper airway obstruction and has a positive correlation to OSA.

B. TREATMENT OF OSA

Treatment of OSA can be grouped into three techniques: behavioral, medical, and surgical. Behavioral techniques include weight loss (where applicable), avoiding sleep deprivation, and avoiding supine (back) sleeping position where OSA is most commonly observed. Avoidance of alcohol and sedatives will decrease OSA occurrences, and smoking should be discontinued

since it can irritate the upper airway mucosa, which can lead to worsening the obstruction (Reite, Ruddy, & Nagel, 2002). Eating large meals before bedtime should also be avoided (Ballard et al., 2007).

When behavioral techniques will not improve OSA, one must turn to medical techniques. While there are no current pharmacological treatments, there are several options that can be prescribed. The most common treatment is in the form of nasal continuous positive air pressure (CPAP) device, and is considered the “gold standard” of apnea treatment devices (Aarab, Lobbezoo, Hamburger, & Naeije, 2011). The CPAP device delivers positive air into the airway by use of a mask (see Figure 2). The amount of pressure delivered by the CPAP machine normally ranges between 6 and 14 cmH_2O (~600–1400 Pa). The airway is never allowed to collapse when this pressure is delivered, thus allowing freedom of breathing. The air pressure delivered varies from person to person, but the correct amount can be obtained during the polysomnogram.

CPAP has been shown to positively improve breathing, oxygen saturation, cardiac rhythm, sleep quality, and daytime alertness (Reite et al., 2002). CPAP has been found to decrease disease severity and subjective sleepiness within 3 months of treatment (McArdle, Devereux, Heidarnejad, Engelman, Mackey, & Douglas, 1999). CPAP has been used since 1981 for treatment of OSA, and it appears to have little known risk associated. While little risk is associated, CPAP does have issues. Compliance, or continued use, is roughly 60 percent for those prescribed (Ballard et al., 2007; Reeves-Hoche, 1994). Compliance issues include lack of sleeping positions (mask can lose air-lock seal on stomach and side sleeping), nasal dryness, mask strap irritation, claustrophobia, and breathing discomfort caused by overwhelming positive air pressure directed into the mask by the CPAP machine.

Figure 2. Nasal CPAP Devices (from “TrueBlue,” n.d.).

People who cannot use the CPAP device can be prescribed a removable oral appliance called a mandibular advancement device (MAD). Oral devices are generally for those with milder forms of apnea and are considered as an effective alternative to CPAP (Johnson et al., 2003). The MAD device is placed in the mouth and moves the lower jaw (mandible) forward, thus moving the tongue forward enough to prevent it from collapsing into the upper airways or pharyngeal airspace during sleep (see Figure 3). The amount of protrusion needed will vary from person to person, and requires much trial and error (Haskell et al., 2009). There is no direct correlation shown between amount of mandible advancement and efficacy of the device, but most professionals believe the initial starting point is one-half to three-fourths of the maximum range of jaw protrusion (Gagnadoux, Fleury, Vielle, Petelle, Meslier, & N'Guyen, 2009; George, 2001). The mandibular advancement range is usually between 5 and 19mm.

MAD is not without its own compliance issues. MAD is associated with jaw discomfort, dry mouth, and dental pain—mostly due to the extreme mandibular advancement that is required for maximum efficacy of the MAD appliance to allow for the desired airway patency. Since the MAD appliance uses the teeth to move the jaw forward, it has been reported that this force can cause a change in the angulation of the teeth, especially the mandibular and maxillary incisors (Gagnadoux et al., 2009).

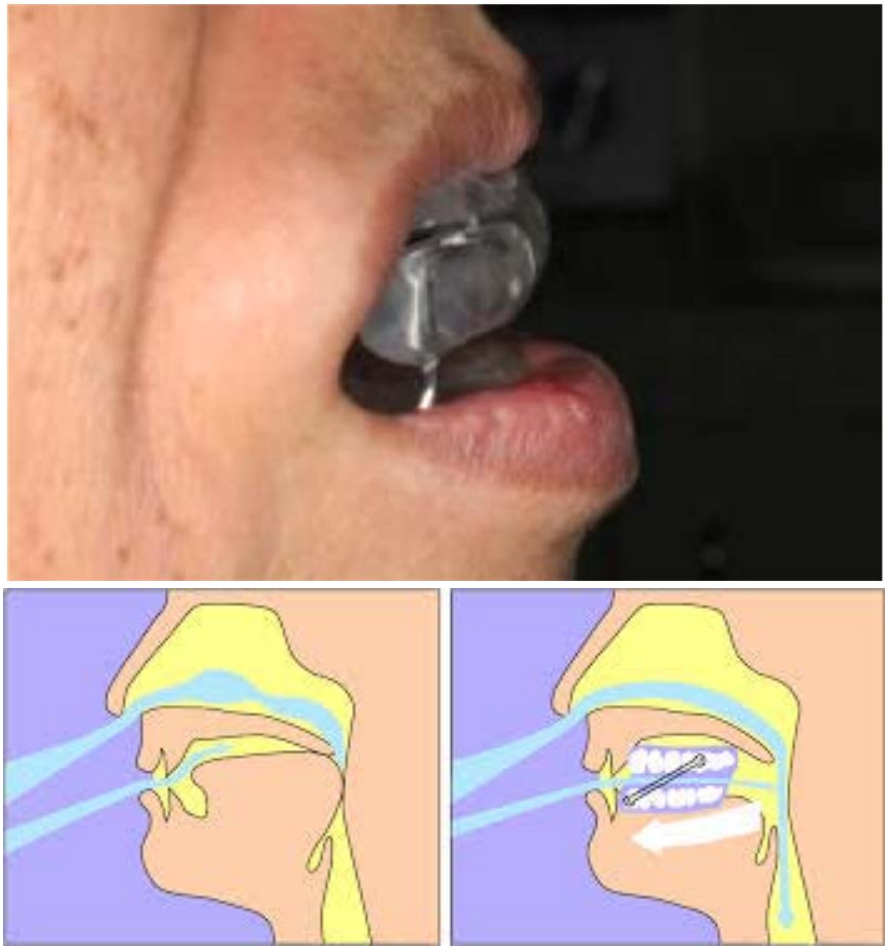


Figure 3. Mandibular Advancing Device (after Schlaflabor-Saletu, n.d.).

Lastly, when behavioral and medical techniques have been exhausted the last OSA reduction technique is surgery. Within the last 25 years, the most reliable surgical treatment has been Tracheostomy, which bypassed the blockage by placing a hole in the windpipe. This procedure is rarely

performed now due to more modern techniques (Thorpy & Yager, 2001). Uvulopalatopharyngoplasty (UPPP) is a technique that shortens the soft palate at the back of the throat (see Figure 4). This procedure does not relieve the obstruction, but removes the soft tissues that vibrate during snoring and can contribute to airway restriction. UPPP has unpredictable results, and research has shown the success rates are approximately 50 percent (Reite et al., 2002). Laser-assisted uvulopalatopharyngoplasty (LAUP) is a procedure that requires no cutting, but rather, uses directed laser that cause the tissue to die. If this procedure is successful after several applications it can cause the soft palate to shrink, reducing snoring. More extensive surgery techniques include Mandibular relocation or procedures to remove the obstruction at the base of the tongue (Thorpy & Yager, 2001). Morbidly obese patients are often directed to reduce body weight, which can be achieved by gastroplasty or stomach surgery, when normal weight loss attempts have failed.

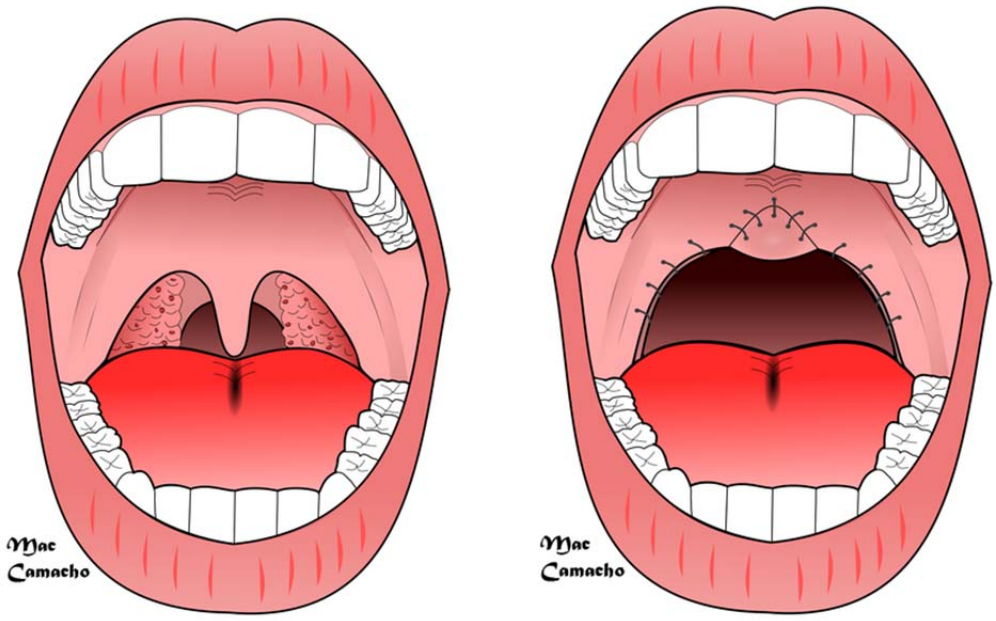


Figure 4. Uvulopalatopharyngoplasty (before and after surgery) (from Comacho, 2014).

C. OSA IN MILITARY/DOD

From 2008 to 2010, the number of United States veterans receiving disability benefits for sleep apnea increased by 61 percent, from 39,145 cases in 2008 to 63,118 cases in 2010 (Brook, 2010). It is estimated that one out of five war veterans suffer from sleep apnea, conversely only five out of 100 people in the general population have been diagnosed with OSA (Brook, 2010).

OSA has been evident in the military for a long time but not diagnosed in service members until the mid-1970s. During WWII, it was recommended to Soldiers who were loud snorers to wear their rucksack during sleep to prevent them from sleeping on their backs (snore-prone), thus preventing the snoring from giving away their tactical position (Johnson et al., 2003).

Other than being purely a tactical nuisance, OSA causes many side-effects that concern military readiness and total fitness. Sleep deprivation is known to affect abilities to handle stress, cognitive skills, relationship skills, and physical conditioning that keep warfighters ready for battle. The Human Performance Resource Center (HPRC), which is a Department of Defense (DOD) initiative under the Force Health Protection and Readiness Program, states that most Soldiers needs between seven and eight hours of sleep to perform their duties optimally (Seelig, Jacobson, Smith, & Hooper, 2010). Warfighters who have OSA can never get that much quality sleep, and the body cannot be trained to need less sleep than the normal baseline amount. According to HPRC, “with total sleep deprivation, performance typically declines by 25% every 24 hours (depending on the type of performance) being measured”. Of concern to DOD, is recent studies have shown that warfighters who have had combat experience are significantly more associated with having trouble sleeping than those who have not been deployed (Seelig et al., 2010).

A recent study performed from Madigan Army Medical Center, Joint Base Lewis-McChord showed that 725 active duty military personnel who complained of having a sleep disorder were diagnosed with mild OSA (27 percent), insomnia

(25 percent), moderate-to severe OSA (24 percent) and paradoxical insomnia (five percent) (Mysliwiec, McGraw, Pierce, Smith, Trapp & Roth, 2013). Also troubling is that the researchers found that the average amount of time the Soldiers spent sleeping each night was only 5.74 hours, far below the recommended seven to eight from the HPRC. According to Lieutenant Colonel (Lt. Col.) Vincent Mysliwiec, a researcher in the study,

> *When Soldiers don't get enough sleep and are performing hazardous duties, there is a greater chance for life-threatening error. Sleep deprivation over time is also associated with major health problems such as chronic pain, cardiovascular disease, obesity and depression. Sleepiness is also the second leading cause of traffic accidents—akin to being legally intoxicated*. (Resweber, 2011)

In 2011, the Army attempted surgical treatment on 37 Soldiers diagnosed with moderate-to-severe OSA. The treatment performed was maxilla-mandibular-advancement (MMA) surgery, which increases the diameter of the upper airway. According to the study, titled *Surgical Treatment for Adult Obstructive Sleep Apnea: A Systematic Review of High-Level Evidence*, six in 10 Soldiers (or 59 percent) reduced their apnea-hypopnea index (AHI) by at least 50 percent ("Army Physicians," 2011).

The cost of untreated OSA has been shown to have a dramatic impact when it includes the sequelae or pathological conditions resulting from the disease. Studies have found that the mean medical costs of OSA sufferers, prior to being diagnosed with OSA, were approximately twice what was seen in sex and age matched control subjects (Kapur, Blough, Sandblom, Hert, de Maine, & Sullivan, 1999). This study, performed by the University of Washington, pointed out two implications. First, untreated OSA may be a factor in total medical expenditure in the U.S. due to a large increase in medical costs of the undiagnosed. This study has estimated medical cost attributed to untreated OSA having a burden of $3.4 billion per year (in 1999 dollars). Secondly, it also found that there could be a significant reduction in costs treating adverse sequelae of

OSA by the offset costs in treatment of OSA. So finding patients suffering from OSA, and treating them is a good measure in reduction of long-term costs.

Currently, the DOD and the Veterans Association (VA) is covering the cost of testing and OSA equipment, at a cost of approximately $500 million a year for treating veterans (Philpott, 2013). While this is a good step in reducing long-term health costs, our study is seeking ways to ensure that Soldiers adhere to the treatment by improving compliance. As prior studies have shown, only 60 percent of patients continue to use their prescribed OSA device (Ballard et al., 2007). Noncompliance will result in increased health costs due to higher levels of depression, coronary artery disease, hypertension, etc. that are known symptoms association of those diagnosed with OSA.

The implications mentioned above drive our study. We have an earnest appreciation for the current devices and techniques available to treat OSA. What we intend to do is use modeling and simulation techniques to improve readily available devices by addressing the compliance issues and using state-of-the-art techniques to field them. Our end state is to improve Soldier readiness, while also creating a cost benefit for the DOD.

III. METHODOLOGY

A. MODELS

1. Healthcare Simulation Model Review

Modern healthcare and medical simulation tools are mainly developed and used for education and training. It is very understandable why. In 2010, the Office of Inspector General for Health and Human Services attributed 180,000 deaths to human error and bad hospital care (Allen, 2013). While a recent study estimates it could be at least 210,000, and up to 440,000 patients who die every year from hospital mistakes (Allen, 2013). This would make hospital error the third leading cause for deaths in America, just behind heart disease and cancer.

While healthcare and medical simulations are needed for training of healthcare workers and doctors, we want to explore how modeling and simulations are being used for medical device design, testing, and evaluation. Specifically, what tools are out there? How can they incorporate to be used in medical device development and engineering? Can they assist or improve live human testing, or result in the lack of necessity for live testing? Where are improvements in future modeling and simulations for healthcare needed?

a. Santos

Santos is a virtual human avatar that can be used to simulate human motion and predictive posturing (see Figure 5) ("SantosHuman™ Inc.," 2013). In order to make Santos realistic, the developers used biomechanics, physics optimizations and clinical evaluation. Santos can play a large role in evaluating the manufacturing feasibility and safety concerns prior to making costly prototypes.

Santos has real value in assisting with the engineering and development of new military applications and platforms by enhancing ergonomics and fatigue relief by adjusting the "synthetic" environment and products before "real" product

production. The Santos avatar could be used in medical advancement due to its high fidelity, biomechanically accurate model of a person, including the physics of bone and muscle.

Figure 5. Santos predictive dynamics capabilities (from SantosHuman™ Inc, 2013).

Because Santos combines a biomechanical musculoskeletal model along with predictive dynamics technology, it can be used to predict motion and will react mechanically like a human. Santos can give the engineer feedback on how a certain task or combinations of movements will impact human fatigue, speed, strength and torque over a period of time (Stackpole, 2011).

While the Santos human model involves bone and muscle physics, it is limited with regard to internal organ representation. Where Santos has focused development of the human hand, developing a model with 25-degrees of freedom and including muscles and tendons, it has not done the same for the pharyngeal airspace or lungs. The Santos model does give hope, within the ever exponentially increasing speed of technology, that we will have a human model that is capable of accurate inter-organ product design prior to prototype development.

b. Zygote

Originally developed by Google engineers as Google Body, the Zygote human model is an accurate 3D model of the human anatomy. The Zygote website claims that their anatomy models are the most accurate, detailed, and comprehensive models available (“3D Human Anatomy," 2013). The Zygote model allows the user to dissect the human tissue for deeper views into the anatomy down to organs, veins, and nervous system (see Figure 7).

Zygote allows computer-aided design (CAD) engineers to use the models for development of medical devices and products that interface with the human anatomy. The 3D models are high quality polygon-rendered models that support many major formats, such as Maya, 3D Studio Max, Lightwave and Softimage. Some of the detailed models available for engineers are the human anatomy model, muscular skeletal connective model, heart model, skeleton model, hand model, respiratory model (see Figure 8), organ model, and circulatory model. The Zygote models are all solid models, using non-uniform rational basis spline (NURBS) construction (see Figure 6).

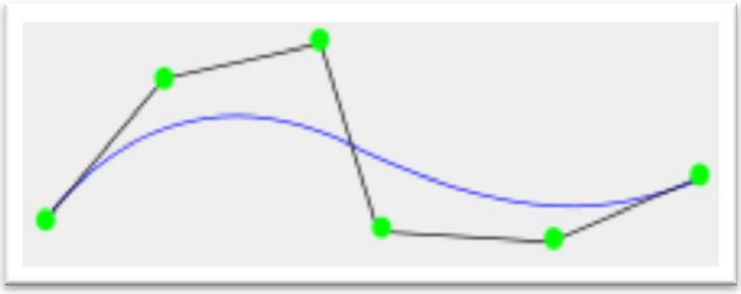


Figure 6. NURBS curve (from *Wikipedia,* n.d.).

NURBS is the current industry standard for designing and modeling surfaces, especially those with complex curves. NURBS are mathematical models that are efficient to calculate and offer smooth approximations of complex surfaces. The NURBS curves and surfaces are similar to Bezier curves, but the main difference is the weighting of the control points. Bezier curves evolve into one direction while NURBS evolve into two parametric directions (“Non-uniform Rational B-spline” 2013).

While the Zygote human model represents the state of the art anatomical model, it does so with a polygonal mesh that does not contain any physics-based movement or computations. This poses problems when using the model for development of medical devices, as in our case. Optimally, a model that combines the biomechanical aspects of Santos with the anatomical accuracy of the Zygote geometry would be preferred for the purposes of this work.

Figure 7. Zygote human models (from ”3D Human,” n.d.).

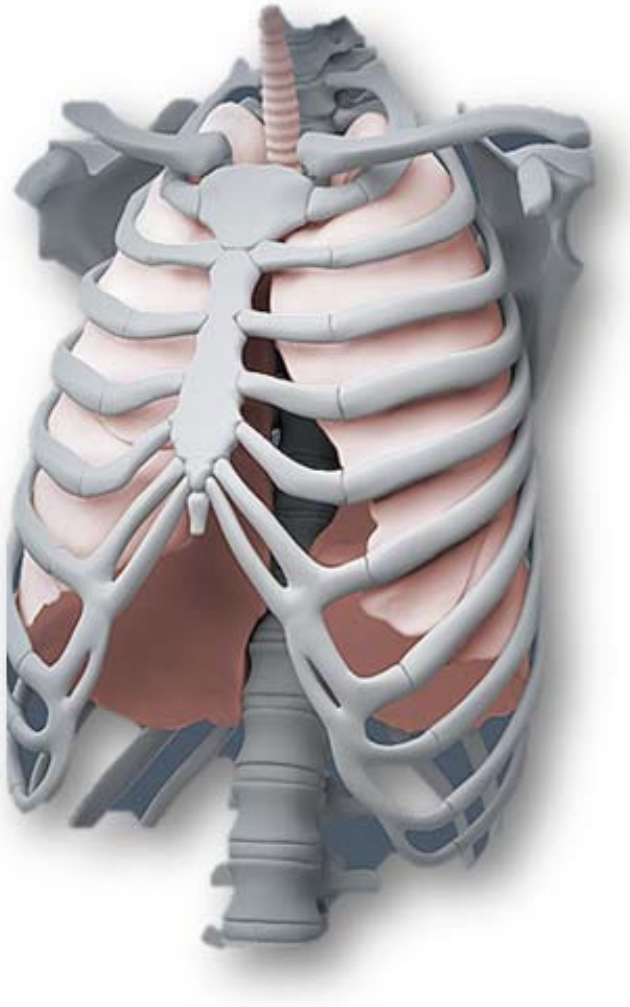


Figure 8. Zygote respiratory model (from “3D Human,” n.d.).

c. ArtiSynth

ArtiSynth is a 3D biomechanical modeling program for physical simulation of anatomical structures (see Figure 9). The ArtiSynth program allows users to create and simulate dynamic mechanical models that have deformable bodies, joints, constraints, and various force actuators (muscles) (“ArtiSynth,” n.d.).

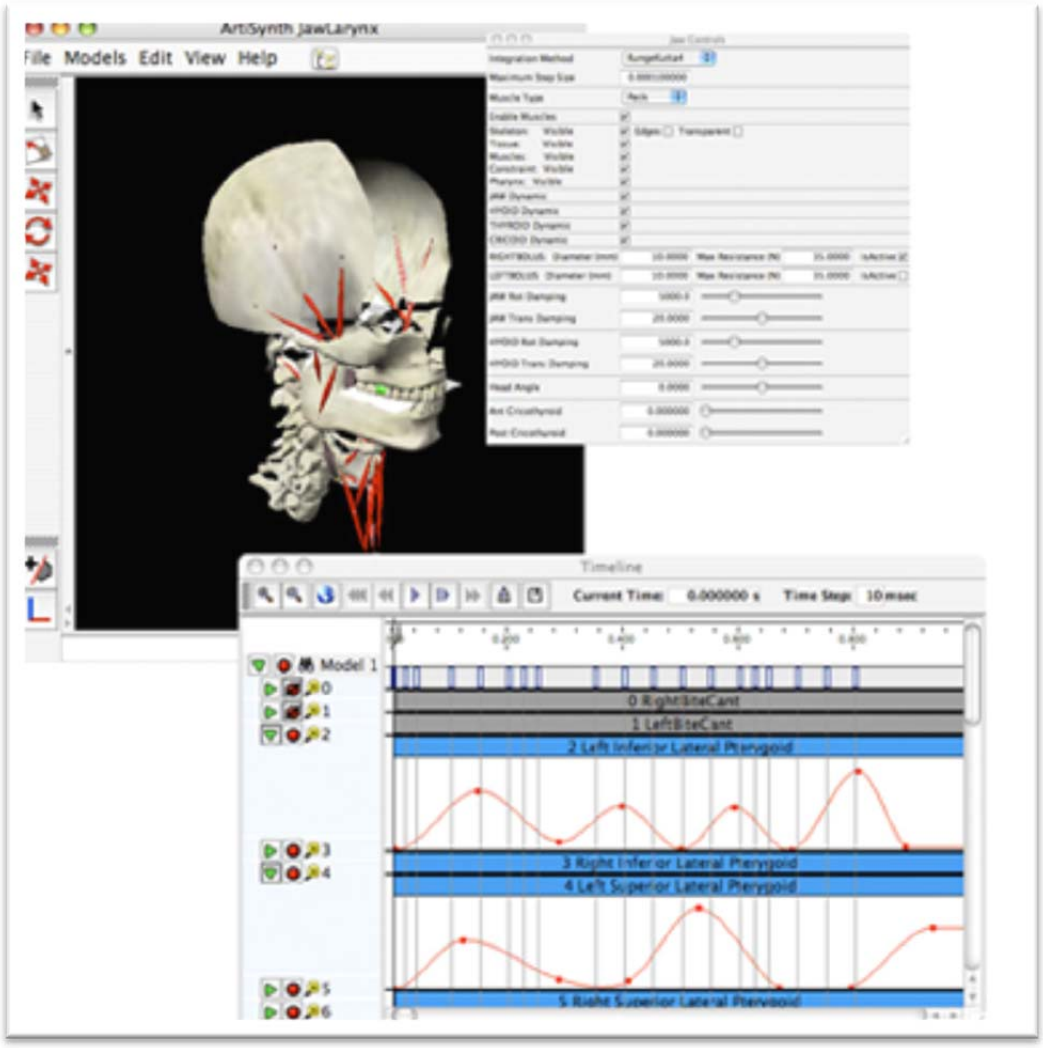


Figure 9. ArtiSynth Modeling Toolkit (from “ArtiSynth,” n.d.).

ArtiSynth has several advantages:

1. General purpose physics engine that combines both rigid and finite element mesh (FEM) based deformable bodies, with constraints and collisions
2. Java-based API
3. Biomechanics support for muscles, inverse computations, etc.
4. Large existing models of the upper airway, tongue, jaw, and vocal tract
5. Enables a graphical Timeline widget that allows user to observe behavior variations from differing parameters and inputs

Since ArtiSynth has a high level of interactive editing and simulation control, so it can be used in broad application domains. Not only is it feasible for mechanical design, but also medical prosthesis design as well. ArtiSynth is currently working on a study, described in Chapter V, to predict whether a patient with OSA is more likely to have a good response to oral appliances by using an accurate representation of the upper airway of the patient in their program. ArtiSynth will be the simulation program we will focus on during this thesis due to the similar application that it is using in its study.

2. Starling Resistor

The Starling resistor is a biomedical engineering device that can help explain how pharyngeal airspace can collapse during obstructed breathing. The Starling resistor was invented by physiologist Ernest Starling for his work in heart preparation and consisted of an elastic fluid-filled collapsible-tube mounted inside a chamber filled with air (See Figure 10) (Knowlton & Starling, 1912). Using this collapsible tube model, the mechanical determinant of airway through the tube is very similar to the functionality in the human pharynx, or deformable airway.

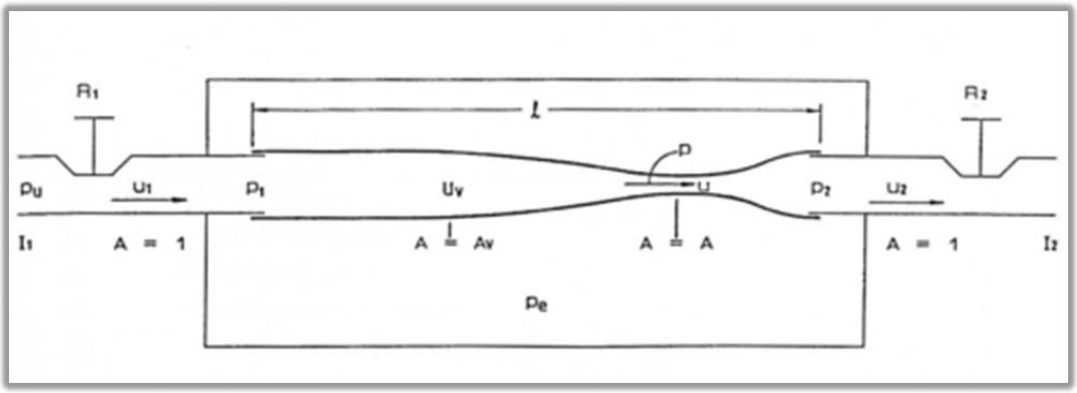


Figure 10. The Starling Resistor (from Armitstead, n.d.).

To describe the Starling resistor as a deformable airway, we must understand how it is made. Two rigid tubes are connected by an elastic tube (L), and a flow of air with volume flux (U_1) is driven through the tubes. P_e is the

outside pressure on the elastic tube, while P_1 and P_2 are the pressures at the upstream and downstream ends of L. During inspiration the upstream pressure (P_u) causes $P_1>P_2>P_e$, and the tube is inflated everywhere. Once P_e starts to overcome P_2 it will cause a collapse at downstream end ($P_1>P_e>P_2$) causing a flow limited state. No flow would be able to pass through the tube as long as $P_e> P_u> P_d$.

A simpler Starling resistor model displaying the upper airway (in Figure 11) shows the critical tissue closing pressure (P_{crit}) in the pharyngeal segment (Mandel & Atkins, 2009). During a flow limited state the maximum inspiration flow is defined by $P_{upstream}- P_{crit}/R_{upstream}$, where $R_{upstream}$ is the upstream resistance.

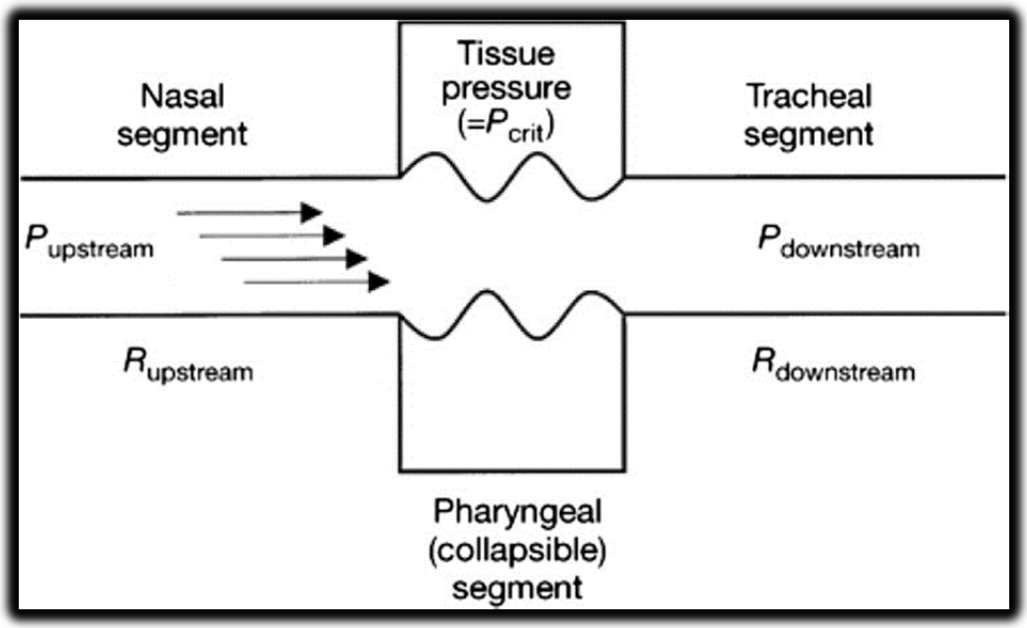


Figure 11. Starling Resistor model of upper airway with collapsible (pharyngeal) segment (from Mandel & Atkins, 2009).

The Starling resistor has two non-linear characteristics (Knowlton & Starling, 1912):

1. "Waterfall effect", where following the collapse, the flow through the airway becomes independent of $P_{downstream}$
2. Self-excited oscillations, such as in snoring

The Starling resistor also explains why males are more likely to be diagnosed with OSA. The gender difference may be related to the longer pharyngeal airway length and the mass of soft tissue that is most commonly found within the soft palate and tongue of males (Dempsey et al. 2010). This phenomenon can also be explained by the deposition of the tendency for males to store fat in the upper body, whereas females typically store fat within their lower body and extremities. However, the Starling resistor does have some limitations in modeling OSA. By increasing pressure $P_{upstream}$ or decreasing $P_{crit,}$ OSA might improve. Furthermore, decreasing $R_{upstream}$ or P_{crit} should increase inspiratory flow, by decreasing the flow resistance (Xu, 2005).

IV. DESIGN

A. MAPPARD ENGINEERING AND DESIGN

Our objective is to utilize a hybrid OSA device, called the Mandibular Advancing Positive Pressure Apnea Remediation Device (MAPPARD), which addresses the main factors/causes of compliance issues with currently available OSA devices. This device was developed and patent-pending by U.S. Army dentist, Lieutenant Colonel (LTC) Michael B. Morehead, DDS. We believe this hybrid oral device can be modeled and simulated using readily available 3D modeling tools that would demonstrate the appliance fit, use, efficacy, and reduction of compliance issues associated with OSA treatment devices, and also give the same or better treatment in a more economical package. Ultimately, our goal is to improve Soldier readiness, while saving DOD long-term treatment costs associated with non-compliance.

Our hypothesis is that using the compliance issues of the CPAP and MAD devices, we can design and model a MAPPARD device that will relieve many of the issues while maintaining the prescribed treatment. As stated earlier, common compliance issues with CPAP include lack of sleeping positions (mask can lose air-lock seal on stomach and side sleeping), nasal dryness, mask strap irritation, claustrophobia and breathing discomfort caused by overwhelming positive pressure. We will attempt to give CPAP patients an alternative sleeping position by removing the mask and using a design that is free from seal leakage and straps. We also think that the amount of continuous positive air pressure can be reduced to a moderate/tolerable level with the addition of an equally tolerable mandibular advancement.

The overall modeling, and design of the MAPPARD device is to use an appliance similar to the MAD, while adding a positive air pressure inlet (in Figure 12). This design allows for manipulation of the jaw while maintaining a proper amount of opening for the air inlet. Orthodontic molds of the teeth will be used to

make custom indentations for the patient to properly seat and wear the appliance during treatment. These custom molds will allow the appliance to have an intimate fit, between the teeth and the device, thus forgoing the need for straps. The seal produced should enable OSA sufferers the opportunity to lie on their sides, in addition to the supine position, and reduce the claustrophobic issues that arise with the mask worn on the external face.

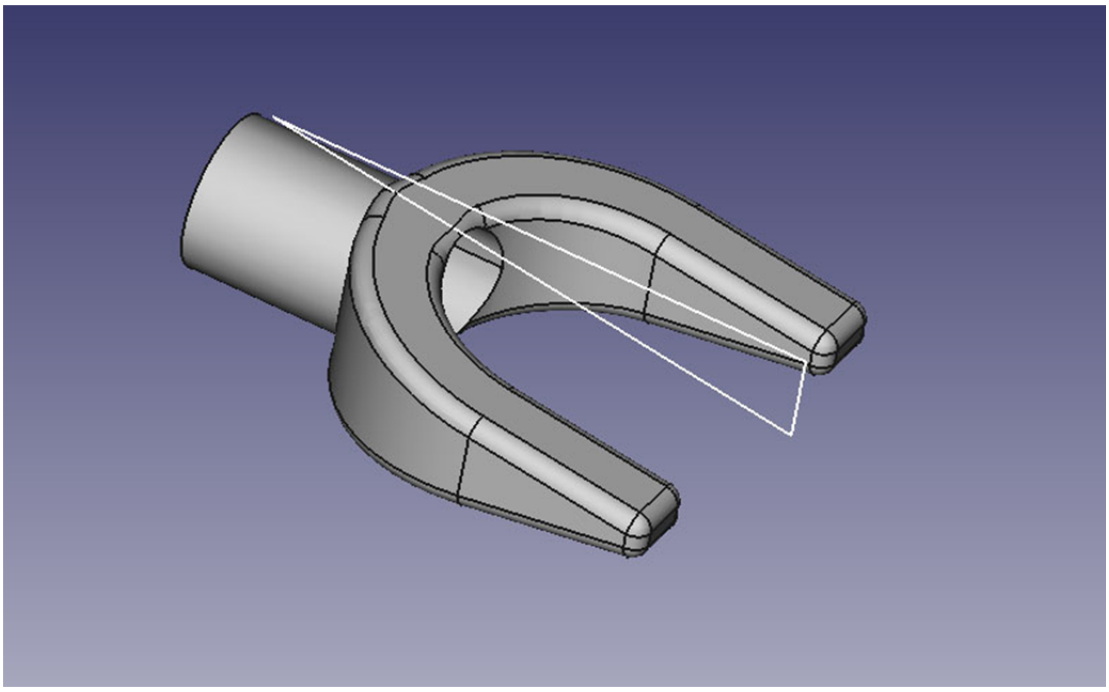


Figure 12. MAPPARD prototype (early design).

For our study, we collaborated with Dr. Morehead for the use and techniques required for fabrication of a mock appliance that will be CAD drawn for accuracy and later manipulated in modeling and simulation rendering programs for our research. Information from a biomechanical study on mandibular advancing devices (George, 2001) was used to determine that the amount of bite opening should be minimized. Opening the mouth allows the anterior attachment of the tongue to swing down, and backwards towards the pharyngeal airspace (see Figure 13).

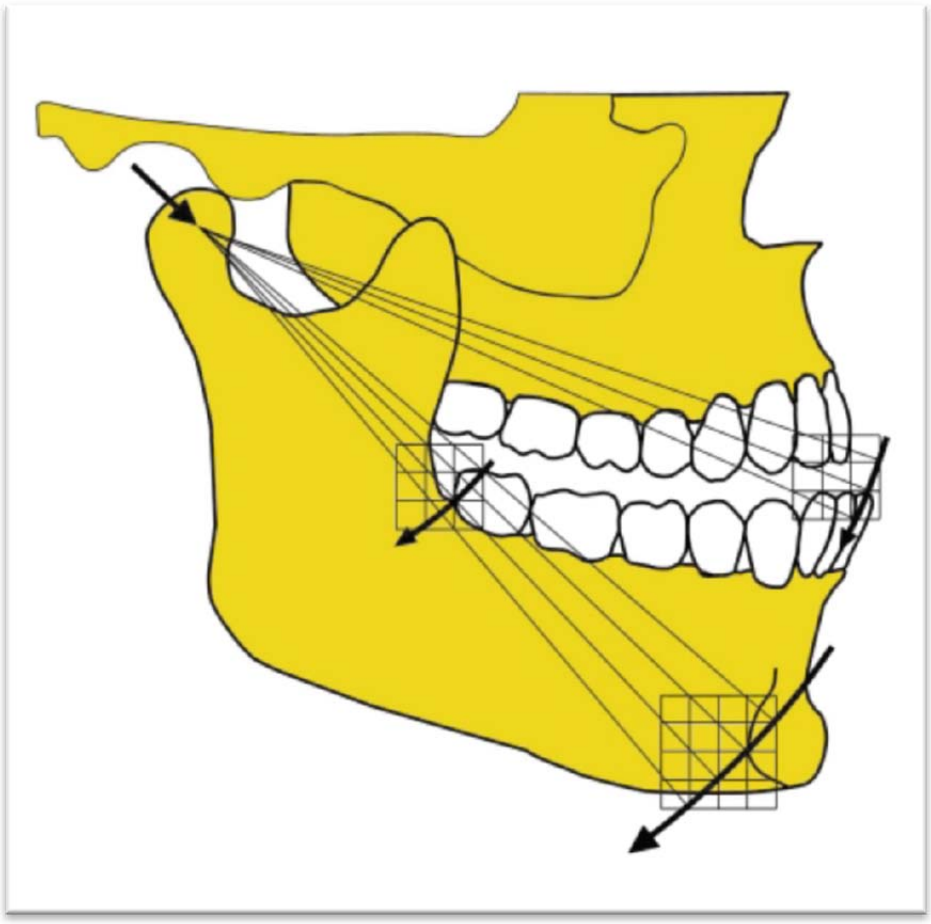


Figure 13. The arrows show that as the mouth is opened the anterior can cause the tongue to move backwards into the airway (from George, 2001, p. 343).

It was found that when using a MAD device the amount of opening was not as considerable due to amount of jaw lateral displacement. In other words, the amount of mandibular displacement will counter the effects of incisor space (jaw opening). Another advantage to opening the jaw is to exert a downward force on the lateral walls of the pharyngeal airspace, thus stretching them longitudinally, which improves the airway by reducing folds and airway resistance (George, 2001). This is important because we plan on using an airway inlet within the opening of the MAD device (see Figure14).

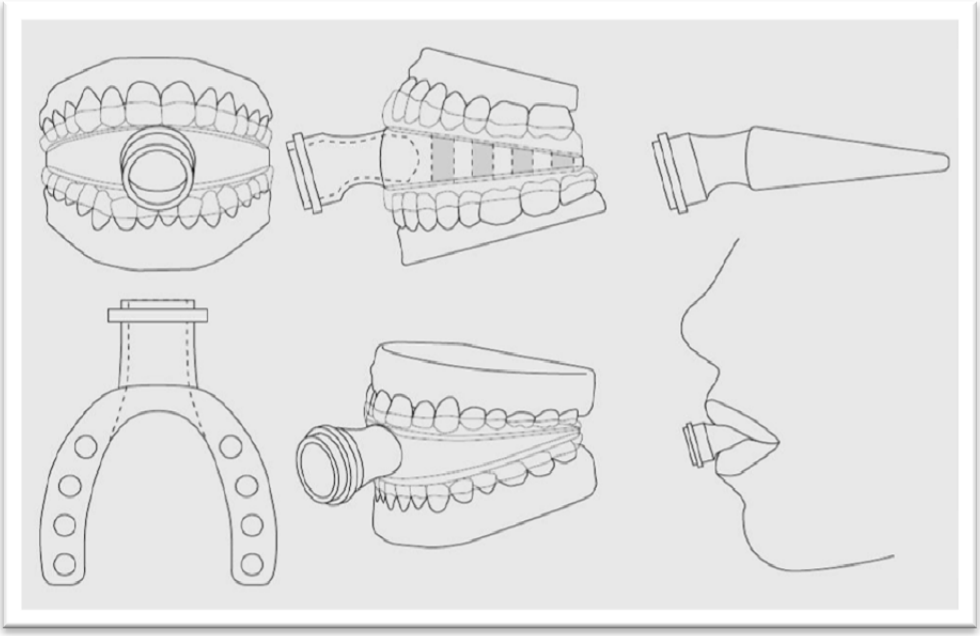


Figure 14. MAPPARD CAD prototype (late design).

Further CAD development of the device shows an improvement on the inlet valve construction, and teeth trays. The MAPPARD device utilizes and inlet valve that will accept a 90-degree swivel and is tapered in the area where the lips intersect for better comfort. Further ergonomic testing will allow us to develop a comfortable inlet valve, roughly 10–14mm of an opening, to allow for standard 12mm i.d. (22mm cuffs) connection to CPAP device hoses.

Several features will need to be determined using computer modeling and simulations. The amount of mandibular advancement will need to be determined, but keeping with our determination to improve compliance, we will seek a moderate level that is comfortable for the patient while also having the benefits associated with MAD. Exactly how much positive air pressure is required to maintain open airway functioning within the pharyngeal airspace will require advanced physics-based models. The amount of correlation between positive air pressure and mandibular advancement will also need to be measured. We will use the ArtiSynth dynamic model to further our research and find solutions to these issues for the MAPPARD device.

B. MAPPARD PRODUCTION TECHNIQUES

Using the most current technology, the best method for producing a physical model of the MAPPARD device economically would be using a 3D dental scanner and 3D printer. Recent technological advancements have allowed dental offices to have alternative materials and techniques to the goopy mold-making of dental impressions. Modern dental devices such as the iTero 3D © digital scanner (see Figure 15) allow dentist to instantly have a 3D digital scan of the patient's teeth and bite. Using a scanning wand, the iTero allows dentists to generate the impression quickly and to capture the image data in real time. The iTero captures 100,000 points of laser light and has perfect focus images of more than 300 focal depths (Lowe, 2012). This device has been seen to have seven times fewer aligner fit issues, based on over 170,000 Invisalign cases using iTero ("iTero," 2013). A device such as this could be used to make the initial digital impressions for an OSA patient that could be sent directly to a laboratory for appliance fabrication, or an in-house 3D printer.

Figure 15. iTero 3D dental scanner (from" iTero," 2013).

3D printers have come a long way in a short amount of time (see Figure 16). 3D printers were first created in 1984, but have recently become inexpensive enough to be considered as a medium for solid object production. Using CAD files, the 3D printer lays small pieces of material until sequential layering builds

up the model. 3D printed models are extremely accurate and have been found to significantly reduce prototyping process time and cost. Once the dental files have been uploaded, the 3D printer will make the teeth models (see Figure 17) that can be used to make the custom splint or dental device. We believe that using a 3D printer will allow rapid development of the MAPPARD device once testing is completed on the design. This technology will allow dental offices to “in-house” custom fit patients that have been diagnosed with OSA, thus relieving the amount of time waiting for the device to be made and also allow for rapid fit and adjustment issues.

Figure 16. Dental 3D printer (from “3DSystems,” 2013).

Figure 17. Printed 3D models (from "3DSystems," 2013).

C. HAPTIC PROTOTYPE

As an initial attempt to reproduce a physical haptic prototype model of our MAPPARD device, we have used the services of Cutting Edge Dental Studio, San Antonio TX. Cutting Edge Dental Studio used a set of dental stone models of a patient's mouth to serve as a test bed for device alignment, fit and visual assistance (see Figure 18). Once the molding process of the dental splints was completed, a semi-flexible bonding material was used to bridge the two splints. A rigid molded tube was inserted between the splints to serve as a connection point for the CPAP machine hose.

Figure 18. Early haptic MAPPARD prototype set into dental stone model.

The results of our first attempt to produce the MAPPARD device proved that the production could be made with standard dental stone impressions (see Figures 19–21). The production also informed the size of the necessary mouthpiece and the requirement for a design for a more ergonomic airway mouthpiece and tube connection.

Figure 19. Early haptic MAPPARD device, side view.

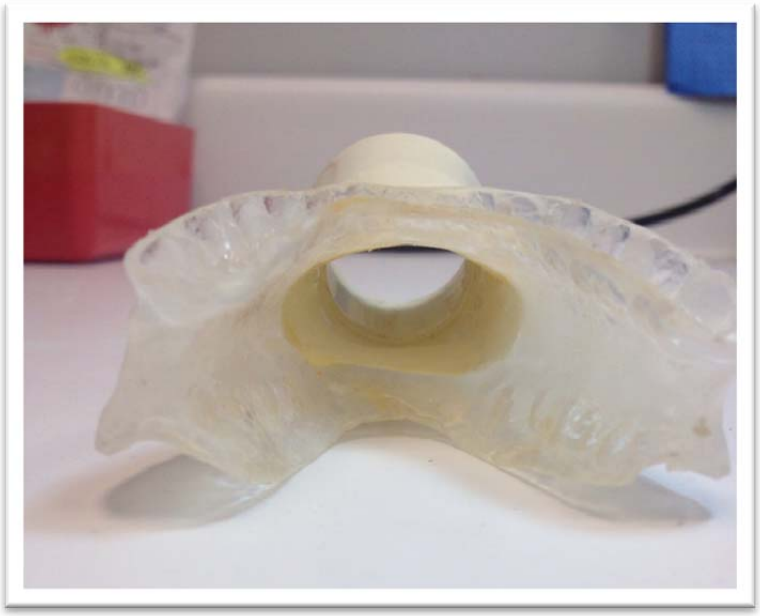


Figure 20. Early haptic MAPPARD, rear view.

Figure 21. Early haptic MAPPARD device with 90-degree swivel.

D. 3D GRAPHICAL MODEL DEVELOPMENT

We decided that the best method for creating, testing and observing the analysis of the MAPPARD device would be greater within 3-dimensional modeling and simulation programs. The first program used was the Blender ©, which is a free, open-source 3D rendering program. Blender was chosen as the graphical modeling software not only because it is free, but also because it uses OpenGL for drawing the interface. OpenGL is considered the industry standard for 2D and 3D graphics due to its cross-language, multi-platform application programming interface (API).

Using the original CAD drawings (Figure 14), the images were layered in the Blender viewport, and separate images were assigned to specific viewing angles (see Figure 22). Once the background images were in place the model of the MAPPARD device could be built to an exact copy within the graphics program. The images allowed for accurate vertices alignment to the edges of the

drawings (see Figure 23). Once the image was built, and the mesh aligned on all angles, it was smoothed and a transparent blue skin was attached (see Figure 24).

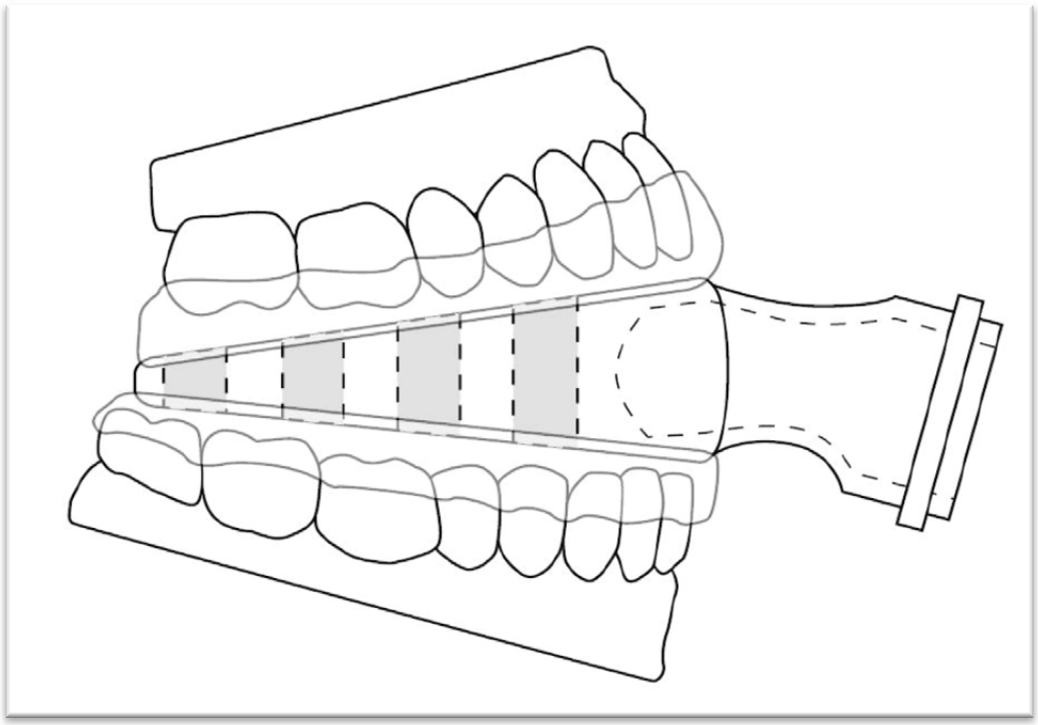


Figure 22. Background image placed in Blender.

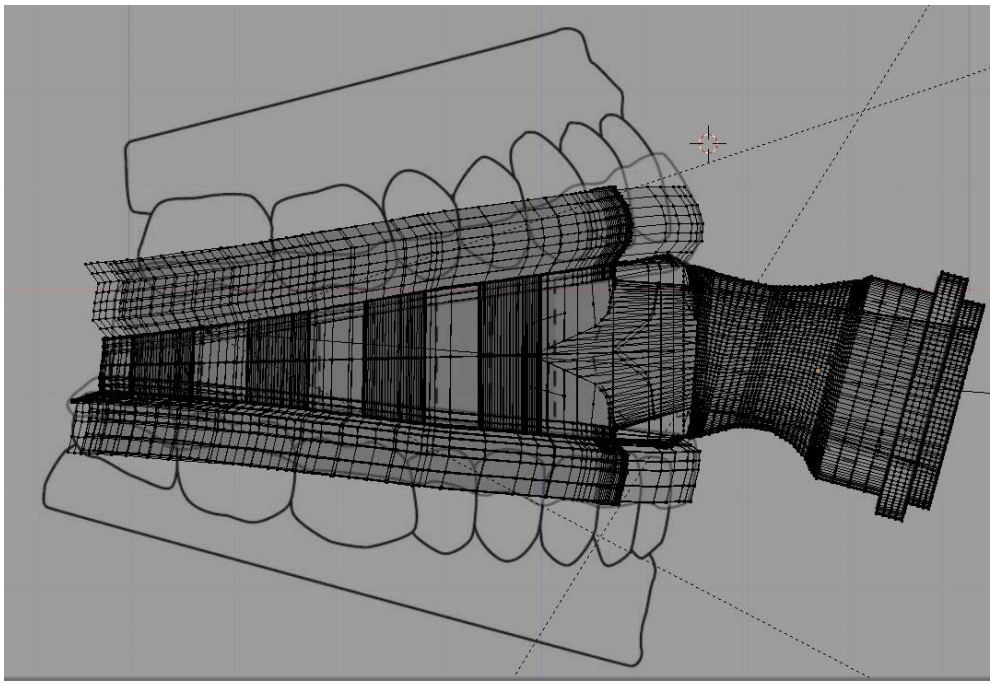


Figure 23. Blender MAPPARD device with vertices rendered.

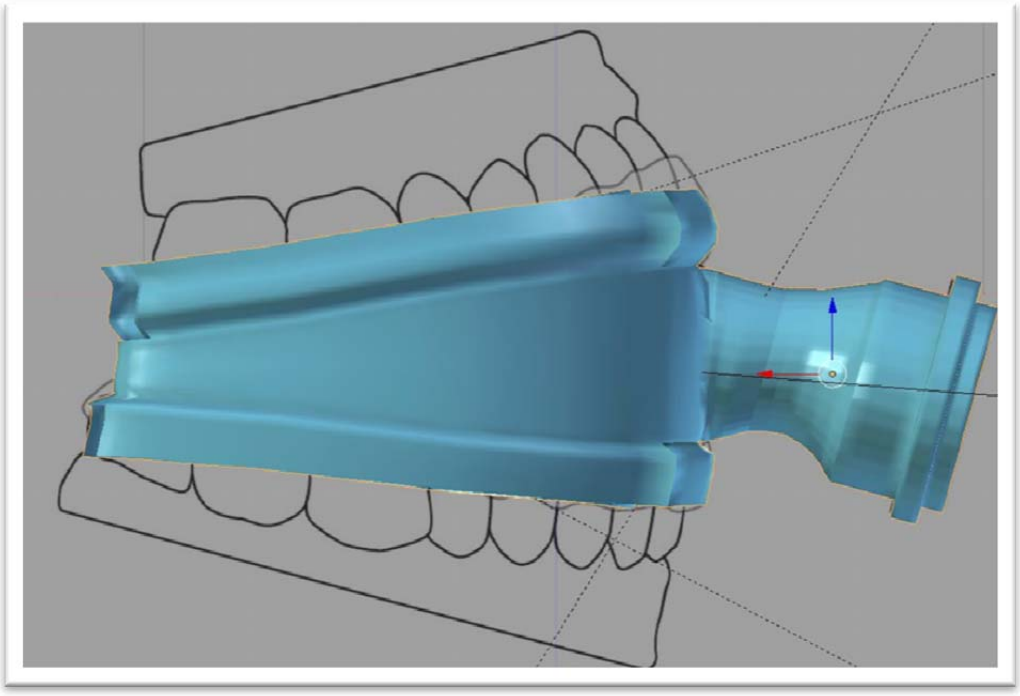


Figure 24. Blender MAPPARD device with transparent blue skin added.

E. MAPPARD DESIGN COST

Estimated cost for an MAPPARD dental device should roughly be the same as a MAD device with the addition of a CPAP machine. Here is an estimated cost for OSA disorder evaluation and treatment ("Cost Helper," 2013):

1. Sleep apnea diagnosis sleep study (Polysomnogram) typically costs **$1,000–$3000 per study**
2. Continuous positive airway pressure (CPAP) machine typically costs **$1,000–$3000**
3. Mandibular device, per Blue Cross and Blue Shield data, averages around **$826** (9–12 months of avg. life of appliance)

Total estimated cost $2826–$6826 for initial evaluation and treatment, with an annual cost of $826 for appliance wear. We believe this falls well within the acceptable treatment cost region. We have used the cost of a typical mandibular device, because we believe the MAPPARD device will roughly have the same cost attributed as the MAD device. Considering the treatment costs associated with non-treatment of OSA, as previously mentioned, and the added benefits of compliance remediation with patients using the MAPPAD device, our device should save DOD thousands of dollars per patient over the lifetime of device usage.

F. 3D PRINTING OF THE MAPPARD MODEL

1. Setup and Slicing

The visual 3D model of the MAPPARD device in Blender® is an effective way to show design points and for a visual display. However, we wanted to be able to show the device prototype in a physical manner and allow for haptic interaction, so we have decided to utilize the Naval Postgraduate School (NPS) 3D printing resources, specifically the Fortus 400mc 3D prototyping printer. The 3D printer allows us to economically achieve these goals and to possibly find deficiencies within our modeling code or design.

The first step for getting our model ready for printing is to use a slicer program. A slicer program takes the 3D model, usually in .STL format, and builds

it into a layer-by-layer machine code (G-code) that the 3D printer needs. A slicer program allows the user to control how the printer uses a print head, or extruder, by altering extrusion speed, head speed, and temperature of the printing media. It also allows for setting other modifications, such as wall thickness and fill patterns/movement.

We have chosen a slicer program cgalled KISSlicer (Keep It Simple Slicer) from http://www.kisslicer.com/. The KISSlicer (see Figure 25) is a free, open-source slicer that allows us to validate our model for 3D printing. While the free version of KISSlicer is limited to a single head printer, the pro version (which requires a donation) allows for setting up the model for multi-head printing.

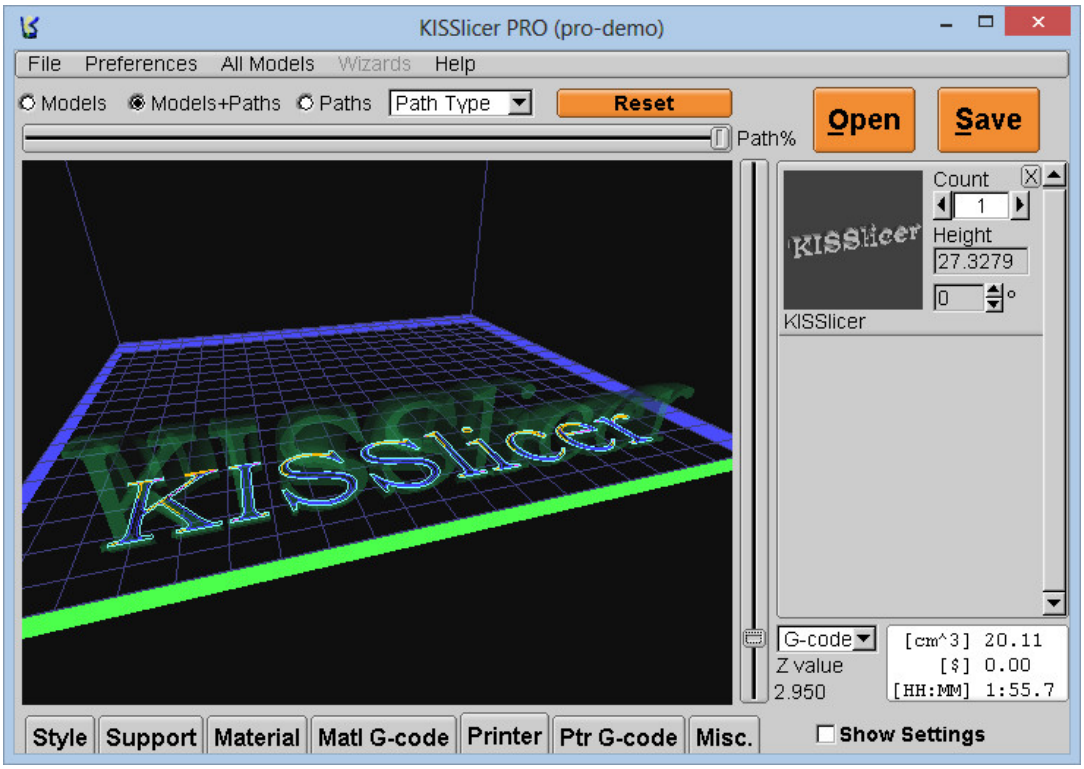


Figure 25. KISSlicer 3D printing software (from “KISSlicer,” n.d.).

Using the KISSlicer, drag and drop the .STL formatted Blender file into the slicer main screen. The first thing is to ensure that the model can be printed. The model must be manifold, or “water tight.” Non-manifold is technically defined as

any vertices, or edge, shared by two or more faces. In Blender, this is accomplished by putting the model in edit mode, deselecting all objects, and enabling the manifold function. Any model vertices that are not manifold will be displayed in orange (see Figure 26). This process is much simpler in the KISSlicer program since it renders the object with the errors automatically displayed (see Figure 27).

Since our model shows some non-manifold errors, we will need to correct these errors prior to submission to the 3D printer. A non-manifold model will not print correctly, or may be rejected by the printer all together.

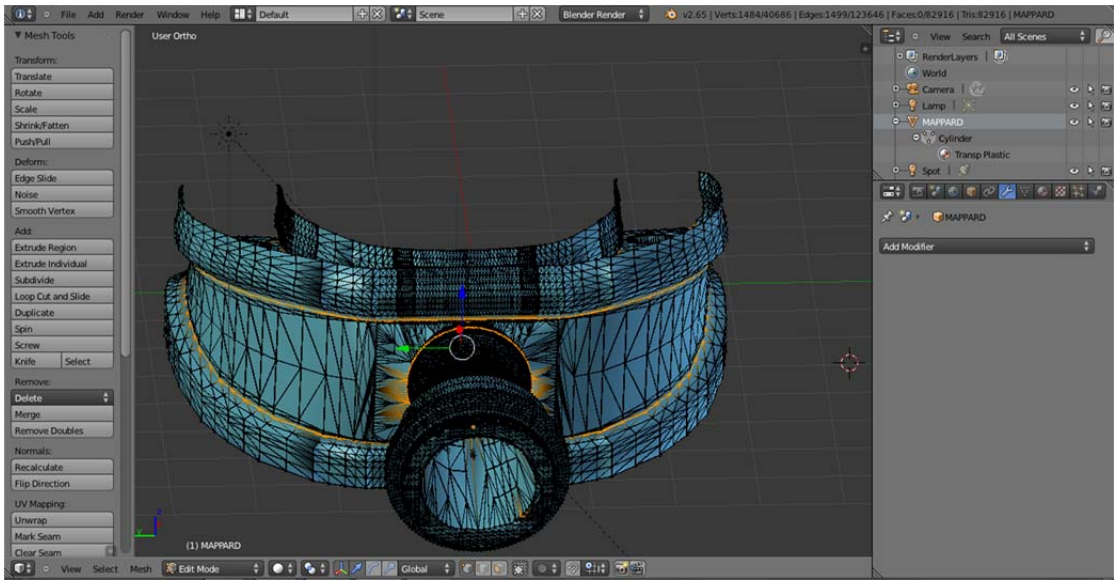


Figure 26. Representation of Blender program highlighting manifold issues in MAPPARD model.

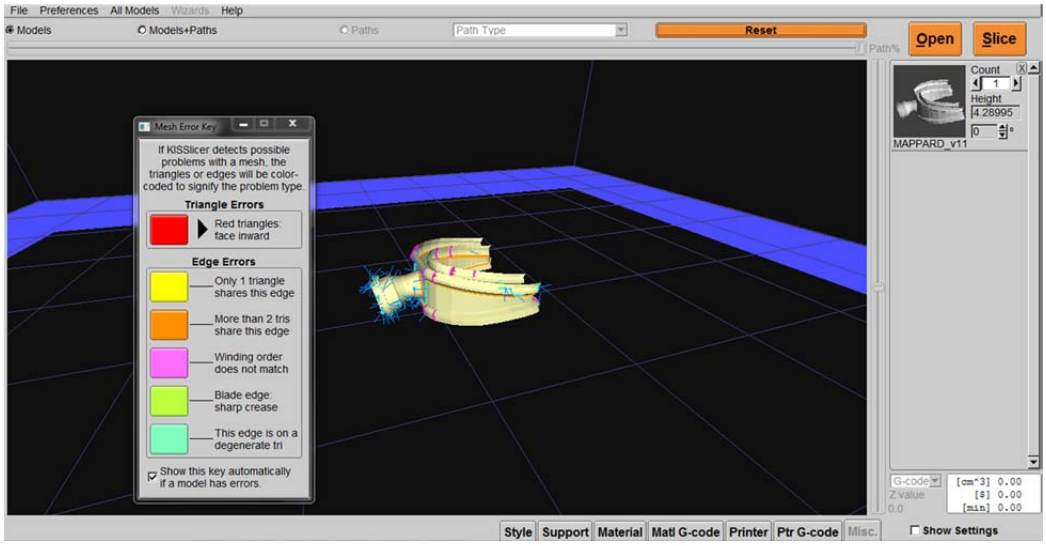


Figure 27. Representation of KISSlicer program displaying manifold issues in MAPPARD model.

When the model is free from manifold errors, it is ready for printing. After preparing the model in KISSlicer, we needed to orientate the model so that the model requires only a small amount of support to brace the model, so it does not collapse upon itself during the printing. The printer will print the support concurrent with our model. Once the printing is completed, the support material can be easily removed.

2. 3D Rapid-Prototype Printing

Computer-aided design (CAD) and 3D printer technology have revolutionized the process of prototyping. 3D mesh models (aka CAD models) can quickly, easily and inexpensively be prototyped. This technique is called 3D-rapid prototyping.

Originally, the use of computer numerically controlled (CNC) milling machines that used the g-code to cut precise “slices” from a solid block of material proved to be effective, but this technique created a lot of excess or wasted material. Another limitation of CNC milling is that intricate or complex designs are not attainable even with the use of modern five-axis milling machines (McGurk, Amis, Potamianos, & Goodger, 1997). 3D printing, as previously

mentioned, solved many of the limitations with CNC milling for use of producing a rapid prototype. By creating a model layer-by-layer, 3D printing can form a skeletonized full-density model with little wasted material, and rather quickly.

Once the setup process was completed, we printed the MAPPARD prototype. The resulting printing of the device is shown in Figures 28–31. The whole process of printing was accomplished within an hour Using the KISSlicer software's estimation tool, the estimated the total cost of printing was less than $0.50 per model printed.

Figure 28. Prototype 3D printing of MAPPARD device with support material (front).

Figure 29. Prototype 3D printing of MAPPARD device with support material (side).

Figure 30. Prototype 3D printing of MAPPARD device without support material (front/side).

Figure 31. Prototype 3D printing of MAPPARD device without support material (rear).

The resulting 3D haptic print of the MAPPARD device, our first attempt, was rather crude, but it showed promise and further enhanced our motivation to continue our research. The scaling of the MAPPARD device within the Blender program and further modified within the KISSlicer program resulted in the author's ability to actually test the model for fit and ergonomics (see Figures. 32 and 33). The subsequent initial test showed that the model was comfortable and without aid of continuous air pressure, easy to breath with the device inserted in the approximate mandibular advanced position.

Figure 32. Trial fitting of 3D printed MAPPARD device prototype.

Figure 33. Trial fitting of 3D printed MAPPARD device prototype.

THIS PAGE INTENTIONALLY LEFT BLANK

V. SIMULATION

A. ARTISYNTH

Testing the MAPPARD design within a simulation is a necessary goal for our research. Testing the design will allow us to validate the model features prior to expensive human-subjects testing. Our previous research in medical simulations for OSA has led us to use ArtiSynth for our study. ArtiSynth, as previously discussed, is an open source biomechanical simulation toolkit that has a Java-based API. ArtiSynth has been developed by researchers at the University of British Columbia (UBC).

A compelling reason to use ArtiSynth is that it allows modeling complex anatomical systems composed of both rigid and deformable geometry. Since the human pharyngeal airway is a very complex system, having the ability to modify the model dynamically and to use ArtiSynth's vast set of components, including rigid bodies, point-to point muscles, and finite elements with both linear and non-linear approaches, provides a suitable simulation environment for our research.

Also important for our research is ArtiSynth's ability to perform physics-based calculations. Specifically, we need the simulation to perform realistic airflow calculations, like seen in Starling resistor. ArtiSynth allows mesh to be created with a modifiable elasticity, or Young's modulus. This will help us to create a realistic pharyngeal airspace that reacts to the addition of airflow as seen in human studies, and allow us to test the MAPPARD device.

B. OPAL PROJECT AND ARTISYNTH

UBC has developed a community of scientists and researchers with the goal of creating a complete biomechanical model of the human Oral, Pharyngeal and Laryngeal (OPAL) complex ("OPAL," n.d.). The OPAL project has been developed to study disorders within the OPAL region by members within computer science, electrical engineering, mechanical engineering, dentistry and linguistics. UBC and the OPAL project researchers use ArtiSynth to gain

understanding of dysfunctions such as OSA, mastication and swallowing disorders, and post-surgical deficits, such as reconstructive jaw surgery.

Since the OPAL project has already performed research and development of OSA within computer simulation, Dr. Sidney Fels, Professor of Electrical and Computer Engineering at UBC, introduced us to research in the OPAL community that seems useful to our purposes here. Professor Fels guidance is appreciated, as he mentioned research not-yet-published conducted by Mr. Peter Anderson, a UBC PhD student. Mr. Anderson has developed a 3D parameterized model of the upper airway and the fluid-structure interaction (FSI) applied to it. His research has potential impact for future work and should be referenced here even though it is not yet published in its entirety.

C. PARAMETERIZED UPPER AIRWAY FSI

The parameterized upper airway FSI model is a 3D model that uses ArtiSynth for running a discrete-event simulation (DES) of the upper airway and pharyngeal airspace (Anderson, 2014). The simulation model uses geometry and mesh that are mathematically defined using flexible parameters, which allow for rapid modification during testing. The model includes both velopharyngeal (upper) and oropharyngeal (lower) cross-sectional areas, as well as tongue and mandible deformable (FEM) bodies (see Figure 34).

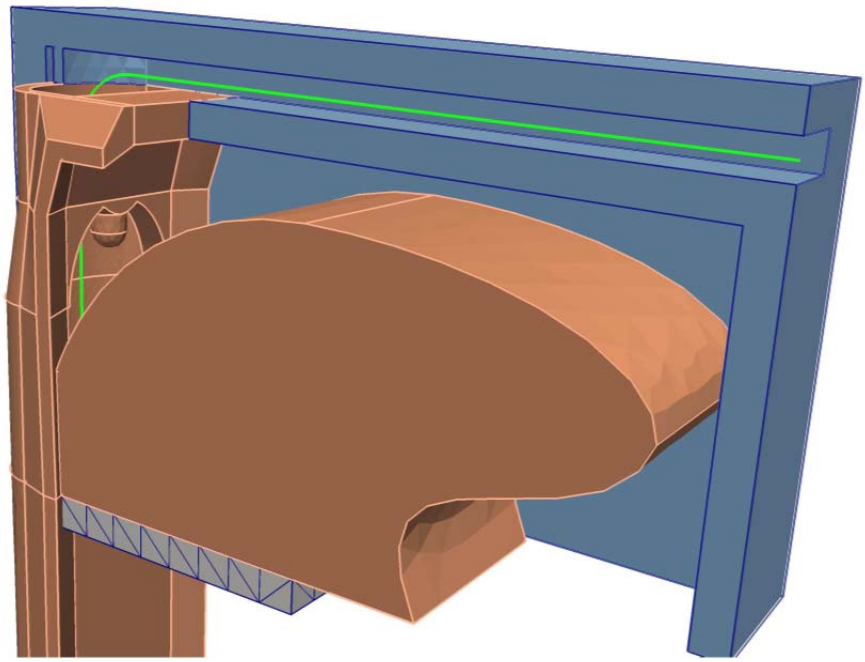


Figure 34. Parameterized upper airway FSI model. The deformable body is shown in pink, while the rigid body is in blue (from Anderson, 2014, p. 43).

The parameterized upper airway FSI model allows for uniform or dynamic airflow pressure, or P_a, to be modified as a positive or negative pressure. The ability of modifying the pressure can replicate normal respiration, or added pressure can be introduced in order to replicate the additional volume from a CPAP machine. Negative uniform pressure will cause the parameterized model to collapse the airway, as seen in the Starling resistor model.

Two measurable areas within the parameterized model that display whether or not the simulation has collapsed the pharyngeal airspace within the upper airway FSI model are the velopharyngeal (VP) and oropharyngeal (OP) cross-sectional areas (see Figure 35). The VP and OP wall thickness are measured at every time-step within the simulation. The resulting measurable data can be used for validating the model against known historical OSA data.

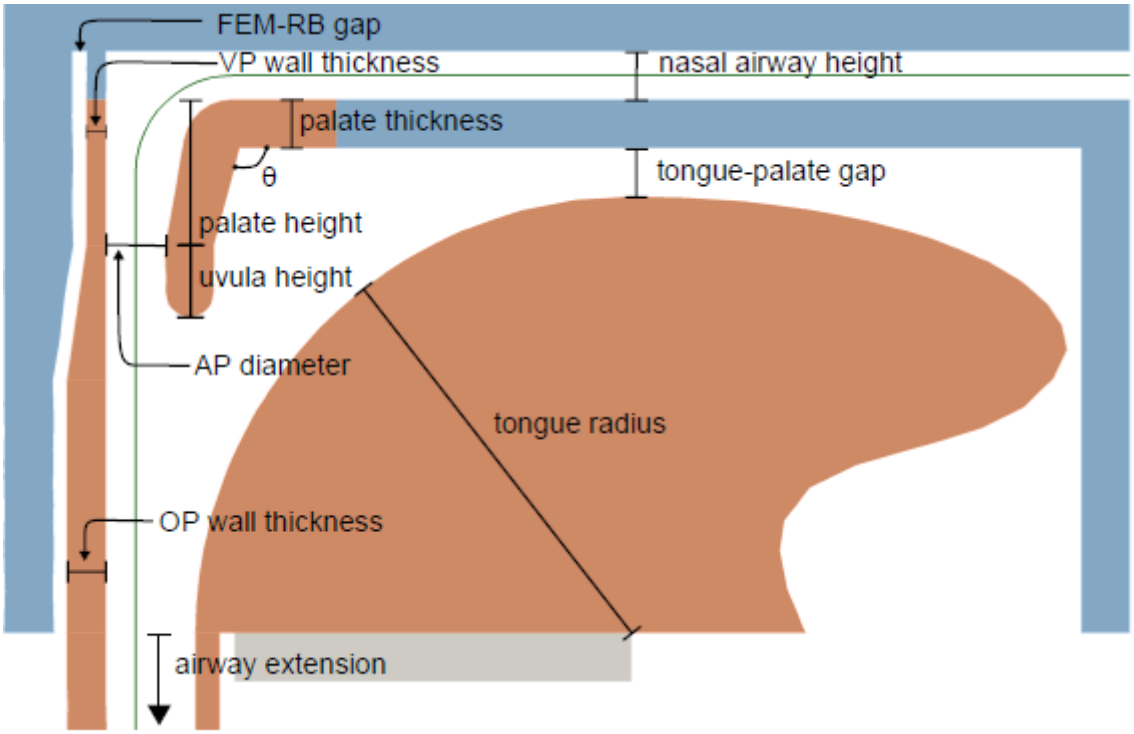


Figure 35. Mid-sagittal view of parameterized upper airway FSI model displaying controllable parameters (from Anderson, 2014, p. 44).

1. FSI Tests of the Parameterized Airway Model

Anderson performed two FSI simulation tests of his parameterized upper airway model: uniform pressure simulations and dynamic flow simulations.

a. Uniform Pressure Simulation

The uniform pressure simulation measures pressure (in Pascal) that is spatially uniform but allows for varying pressure within the time-step of the FSI simulation. The resulting measurable data provided the quasi-steady airway response to pressure and was compared to available clinical measurements.

To demonstrate the model's response to the uniform pressure, Anderson performed a sensitivity analysis using default values initially but incrementally varied one parameter at a time until the model deformed under strong negative pressure (Anderson, 2014, p. 46). The results of the simulation tests concluded

that most of the data showed a fairly linear response, which if validated with MAPPARD parameters should support our testing. He also concluded that the palate of the model made a significant contribution to deformation, and described the airway response as predominantly palate driven. Also noted was that a thinner model has the same response of a softer model, both increased the deformation of the model.

As seen in Figure 36, the FSI model begins with 2000 Pa (Pascal) of positive air pressure resulting in just over 200mm^2 of area. Incrementally the pressure is decreased until the model completely deforms the VP at -2000 Pa, and area roughly 10mm^2 (Anderson, 2014, p. 48).

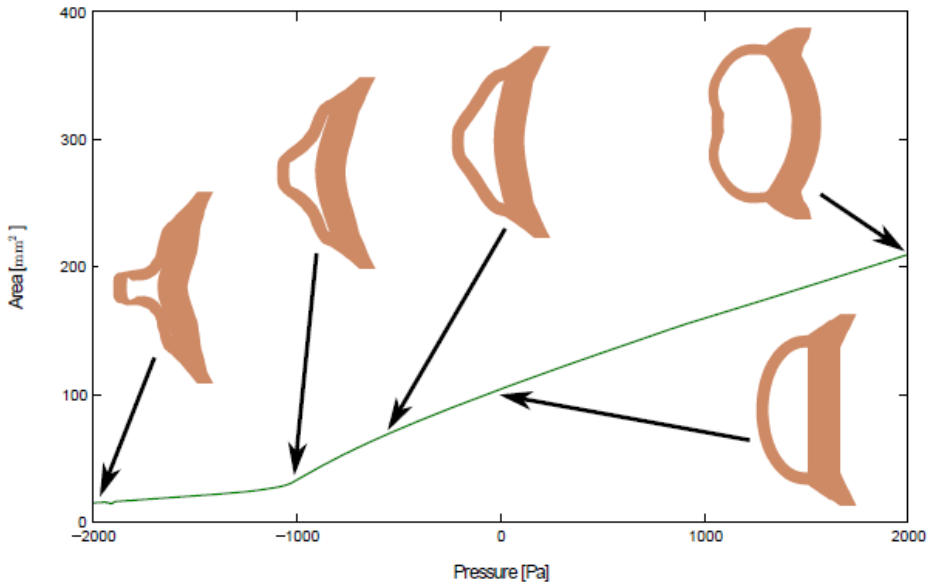


Figure 36. Velopharyngeal cross-sections at important steps within the response of the uniform pressure simulation (from Anderson, 2014, p. 47).

The results of Anderson's uniform pressure simulation were compared to the results from the seminal research of Isono, Remmers, Tanaka, Sho, Sato, and Nishino (1996), called *Anatomy of Pharynx in Patients with Obstructed Sleep Apnea and in Normal Subject,* which proved the model to be accurate in

representing correct collapsing pressure under a uniform pressure. The comparison also correctly predicted diminished airflow availability for parameters associated with OSA.

b. Dynamic Flow Simulation

The dynamic flow simulation used a revised 1D fluid model which allows simulating fluid mechanics and fluid dynamics within a collapsible geometry, Anderson clamped the nasal inlet pressure with a fixed atmospheric pressure, $P_{inlet} = 0\ P_a$ (Anderson, 2014, p. 48). To correctly model inhalation, he applied a sub-atmospheric pressure at the outlet, which is created by the lungs. Simulating the model's behavior for a range of lung pressures, he set the p_{outlet} = -200, -400, -600, -800, -1000 P_a. He then recorded the simulation at a time-step $\Delta t = 0.0005$ seconds for the behavior within $1 \leq t \leq 1.25$ seconds, thus requiring 2500 time-steps for each p_{outlet} used (Anderson, 2014, p. 49).

In this simulation, the pharyngeal wall thickness and palate thickness were varied to find optimal oscillation trends that would react like snoring. The results of this simulation showed a profound effect on the fluid-structure interaction by modifying the flow upstream and downstream of the oscillating region.

The results of the two simulations helped to validate the model against trusted clinical results. According to Anderson:

> by this measure, our parameterized model has good merit, because it predicts numerous clinical observations; collapsing behaviors under uniform pressure, diminished airflow availability for parameters associated with OSA, and oscillations with plausible motions and frequencies for uvular snoring. (2014)

While the dynamic flow simulation proved that the parameterized airway FSI model is fairly accurate at representing the actions of snoring (oscillations), it was deemed not the priority for testing of the MAPPARD device. Our device will not alter the palate, and since pharyngeal wall thickness has been found to act in a predictable and accurate manner, we have chosen to use Anderson's uniform pressure simulation to test the MAPPARD parameters within his parameterized

upper airway FSI model and to compare the results with his clinical measurements. With the uniform pressure simulation, we can modify his parameterized upper airway model to adjust inlet pressure and mandible advancement within a discrete simulation, thus testing a baseline apnea event, as well as, the MAD, CPAP, and MAPPARD devices.

D. MAPPARD SIMULATION

1. Modifying Parameterized Upper Airway FSI for MAPPARD

In order to simulate the MAPPARD model parameters within the parameterized upper airway FSI model (see Figure 37), it was important to modify the simulation to allow the mandible to be controllable on the x-axis. Moving the mandible forward with the code would not only allow use to use the code for the MAPPARD simulations runs, but also could be used for simulating the MAD device as well. As previously discussed, the MAD device range extends between 5 and 19mm. Our goal is to find a more complaint solution to CPAP and MAD, so we will begin with a moderate advancement of 8mm.

The following code was added to move mandible (rbHyoid) forward with each time-step:

```
Point3d rbPos = rbHyoid.getPosition ();
 if(rbPos.x <= 0.008){
rbPos.x = rbPos.x + 0.001;
 }else{
rbPos.x = 0.008;
 };
 Point3d rbNewPos = new Point3d(rbPos.x, rbPos.y, rbPos.z);
 System.out.println(rbNewPos);
rbHyoid.setPosition (rbNewPos);
```

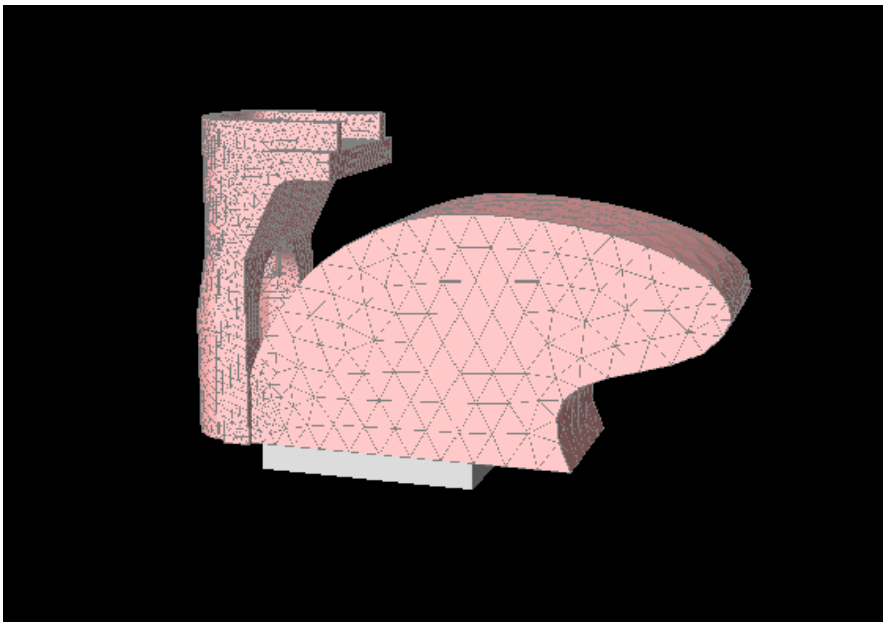


Figure 37. The Parameterized FSI model in ArtiSynth.

Using parameters within the parameterized airway FSI model that Anderson used previously, we initially saw unrealistic measurements when moving the mandible forward with each time-step. While the results were not unacceptable, we deemed it impractical that a device would allow the mandible to be moved forward incrementally during sleep. Our resulting code for moving the mandible would be a fixed amount, just as the design of the MAPPARD device is modeled. For our initial test we moved the mandible to a fixed forward position of +8mm (and later modified to +6, +4 and +2mm), which could represent a MAD device or MAPPARD device (with added positive pressure). We deemed that +8mm would be an acceptable upper limit for advancement. We then could modify to a smaller increment in subsequent tests, if necessary.

2. Testing Parameterized Upper Airway FSI using ArtiSynth

Using the uniform pressure simulation of the parameterized upper airway FSI model we begin with an baseline OSA simulation and compare it to remediation from an mandibular advancement device (MAD) and CPAP machine. Once these baseline results have been determined we can compare

them to our modified FSI model using MAPPARD parameters. Since the FSI model is very robust, we have limited the bulk of our interest within the region that is affected by OSA and where we believe our device will increase airflow: the oropharynx (OP).

The baseline OSA simulation represents the previously mentioned collapsible airway with zero mandibular advancement, and increasing negative air pressure with each time-step until the model deforms at -2000 Pa (50 time-steps). The negative pressure accurately simulates reactions commonly associated with an apnea event, where the OP is collapsed as the tongue and pharynx buckle when the pressure increases beyond the elastic modulus of the pharyngeal airspace.

To model the parameters of a MAD device we used the aforementioned code to advance the mandible. Once the mandible has been moved forward to the specified amount, we incrementally add the negative air pressure that causes the model to deform. Final measurements of the OP cross-section will be captured for comparison testing.

In order to accurately represent the effects of the CPAP remediation device, we need to add positive pressure to the apnea-induced model. As previously mentioned in Chapter II, we know that the normally prescribed amount of CPAP pressure falls between 6 and 14 cmH_2O or roughly 600 to 1400 Pa. For purposes of our test we have chosen to use 1000 Pa, which would fall in the median area of users (Lankford, Proctor, & Richard, 2005).

3. Results and Analysis

We start by presenting the visual evidence of the results of the parameterized upper airway FSI using parameters of the baseline OSA simulation.

As seen in Figure 38, the initialization of the FSI model shows the model from below so we can focus our attention on the OP area. We are concerned

about the airspace that is available during the baseline parameters that create OSA. The airspace is the space that is not filled by the rigid nor deformable mesh, and is measured by the fluid-solid interface. This is due to the fact that the mesh faces feel the pressure predicted from the fluid model.

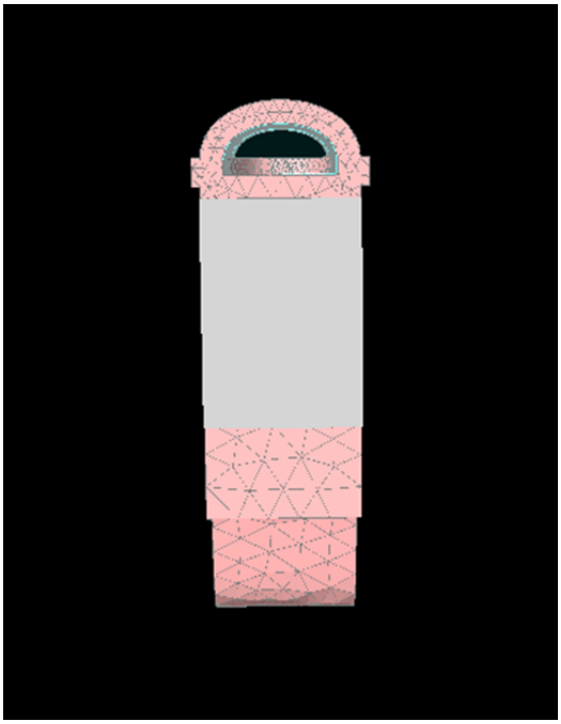


Figure 38. Image of FSI model as seen from below at 0 time-steps.

As the model progresses the negative air pressure increases, airspace is decreased and the deformable bodies collapse. The resulting obstructed apnea event within the pharyngeal airspace can easily be seen within Figures 39–40.

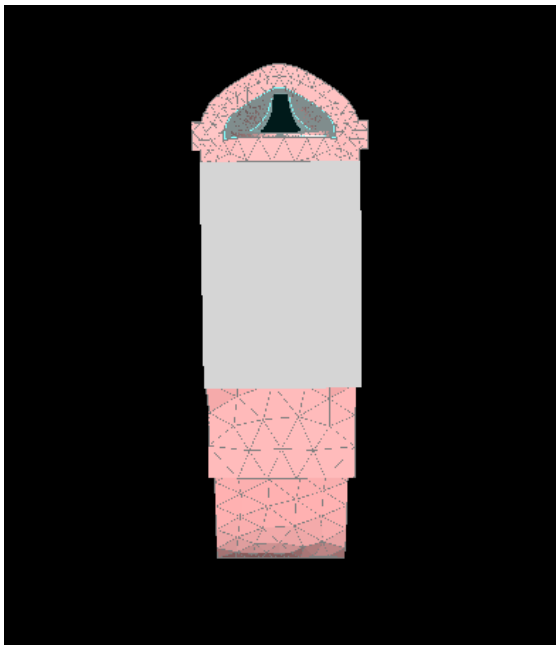


Figure 39. Image of partially collapsed FSI model as seen from below at 25 time-steps during baseline testing of an apnea event.

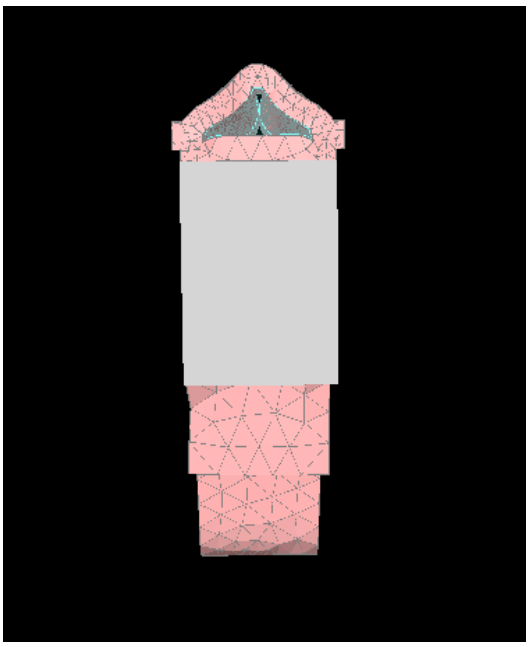


Figure 40. Image of collapsed FSI model as seen from below at steady-state (50+ time-steps) during baseline testing of an OSA apnea event.

At 25 simulation time-steps, we can easily see the effects of the negative air pressure within the model. The model performs as we predicted and further time-steps continue to deform the model as a response to the spatially uniform pressure. While the OP does not fully collapse to zero, it does reach a reliable steady-state area that we can use as a bottom baseline for our analysis. The final airspace volume of the OP is captured for the baseline simulation and further testing is conducted.

Before we begin simulating the MAPPARD device, we completed the testing of the parameterized upper airway model using parameters of the MAD and CPAP remediation devices. The data is preserved for each of the tests, and as seen (in Figure 41) the results of the three tests, using a data analysis program JMP Pro 10.

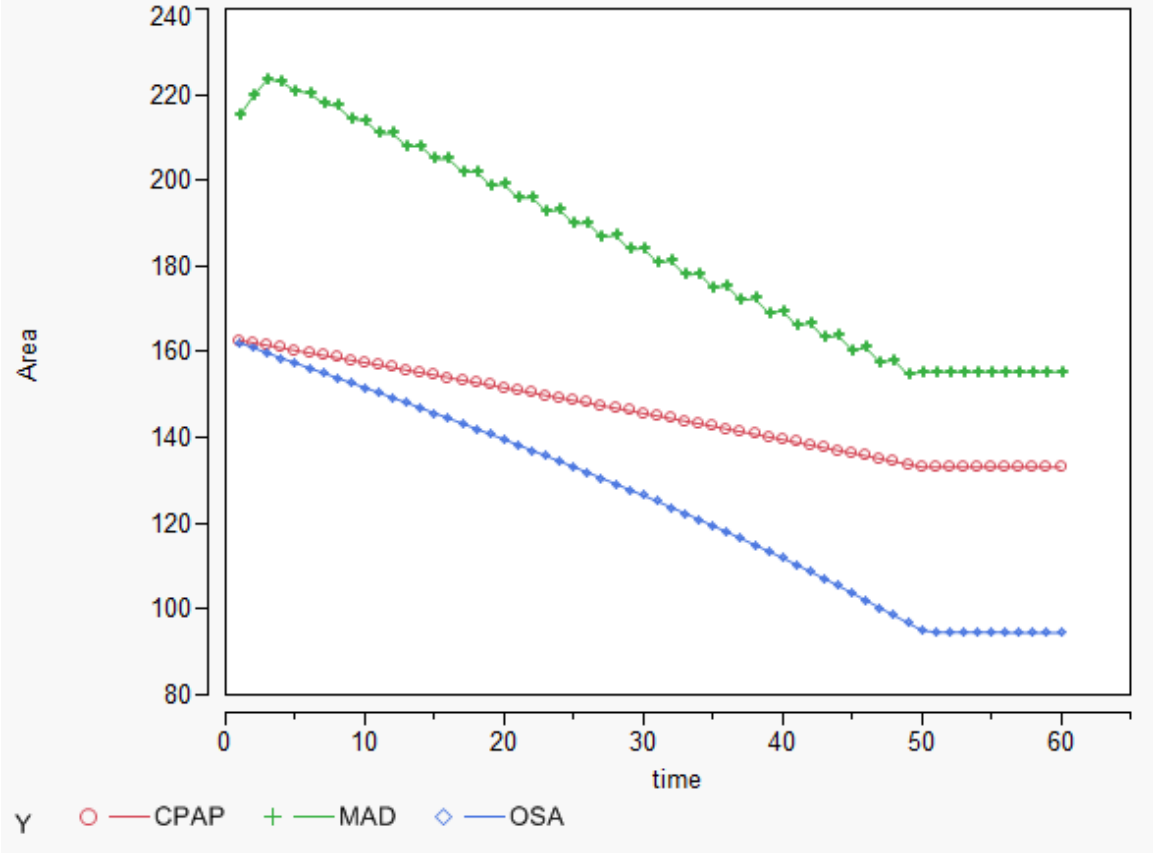


Figure 41. A sequence showing the Oropharyngeal area (mm^2) from beginning of simulation until steady-state occurs using JMP Pro 10.

As we can see, the OSA simulation starts with a cross-sectional area (CSA) of 162 mm^2 and decreases until it reaches steady-state CSA of 95 mm^2. The CPAP simulation also begins with a CSA of 162 mm^2 but remains fairly steady and only decreases by 29 mm^2 to a steady-state CSA of 133 mm^2. Since the MAD simulation requires us to advance the mandible by 8mm, the beginning CSA is rather large at 215 mm^2. However, once the negative pressure is increased the MAD simulation drops the volume of area until it reaches a steady-state of 155 mm^2.

To simulate the parameters of the MAPPARD device, advanced mandibular and added pressure, we want to determine if our model has the same or better performance while being more complaint than the MAD or CPAP. To

accomplish this, we would need to set up the MAPPARD simulation with air pressure less than the CPAP simulation and less advancement than the MAD simulation. We settled on the following MAPPARD test parameters:

Positive Air Pressure (Pa)	Mandibular Advancement (mm)
500	+8
500	+6
500	+4
500	+2

Table 1. Table represents alternate mandible advancement and positive air pressure added during each MAPPARD FSI simulation run.

The positive pressure amount of 500 Pa is half the pressure required for CPAP remediation, and the amount of advancement would be determined by final CSA outcomes. We have concluded that if we can maintain these variables, and if they perform as well as the other two remediation devices, the MAPPARD would show a more compliant solution, supporting our hypothesis.

Each of the MAPPARD simulation tests were run and the amount of advancement was changed by 2 mm for each run. The consolidated outcomes (see Figure 42) display each of the simulation runs as they decrease in area due to deformation from the added negative pressure. The amount of advancement does display a positive increase in area at both start point and at the steady state CSA.

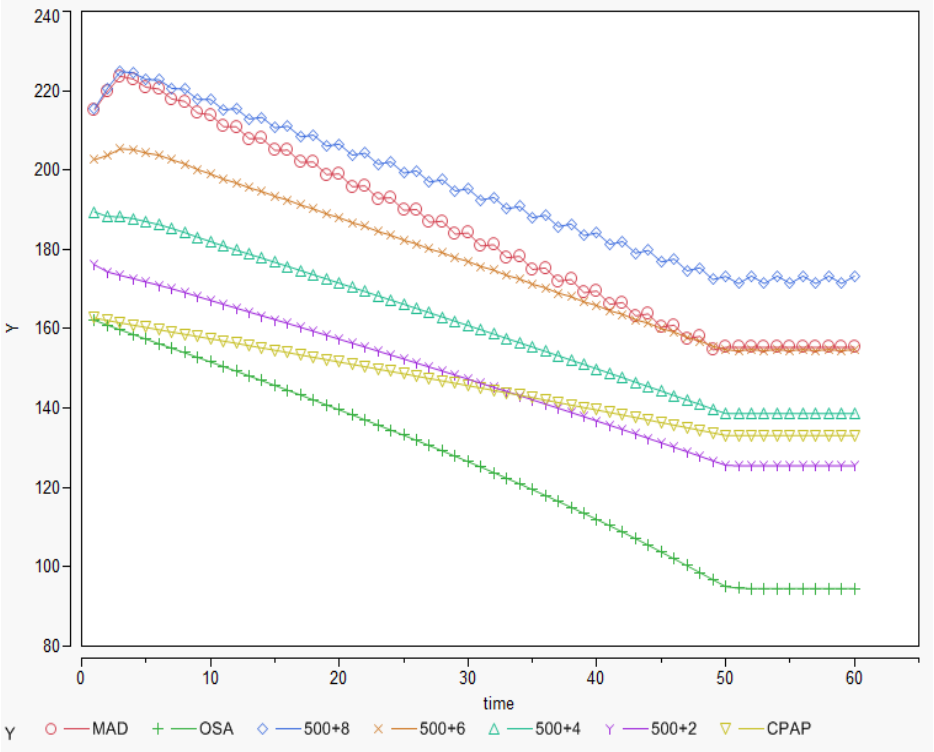


Figure 42. Results showing oropharyngeal cross sectional area (mm^2) from beginning of simulation until steady-state occurred from CPAP, MAD, and MAPPARD (pressure, + mm of mandibular advancement) simulation runs using JMP Pro 10.

If CPAP remediation is the “gold standard” for treatment of OSA (Aarab et al., 2011), than the MAPPARD device performed very well under simulated conditions. Only one MAPPARD parameter did not exceed the CPAP device in final CSA, and only missed the amount by 8 mm^2, which was our lowest amount of mandibular advancement of +2mm.

Looking at the 500+6mm MAPPARD simulation, the CSA possibly displays an optimal output. While being 25 percent (2mm) less advanced than the MAD device and 50 percent (500 Pa) less pressure than the CPAP device, its steady state is even with the MAD device and significantly larger CSA than the CPAP FSI simulation runs.

The CSA of the OP measured at time-step 0, is 25 percent larger (203mm^2 versus 163 mm^2) in the model of the MAPPARD FSI (at 500+6mm) as compared to the OSA FSI model. The airspace result seen in the subsequent time-steps (see Figure 42) are predictably larger in volume within the MAPPARD simulation, and remain larger through the duration of testing. In fact, the volume at 50 time-steps (maximum time-steps required to collapse model) of the MAPPARD device FSI model remains nearly the amount of volume within the baseline OSA FSI model at the starting point (155mm^2 versus 162 mm^2).

The results, while being very promising, leave us with some assumptions that require validation using human-subjects testing methodology. Some of these assumptions are:

- The MAPPARD device is more comfortable than the current apnea remediation devices.
- Combining less air pressure than required for CPAP, and less mandibular advancement creates a more compliant solution.
- The MAPPARD device can be readily made with current dental and industry tools, while also being a cost-effective solution.

E. DISCUSSION AND CONCLUSIONS

Simulating the human upper airway is extremely complex. The complexities arise from dynamic and nonlinear muscle properties, intricate and dynamic geometry, and a large variance between person-to-person pharyngeal space, even varying within the same person depending upon their condition. And while human anatomy is well understood, few efforts have been made to translate that knowledge into simulacra of any value to our purpose.

We set out to find a compliance-increasing solution for obstructed sleep apnea patients, using state of the art modeling and simulation (M&S) tools. While our research has shown that there has been an increase in the use of modeling and simulation for studying human physiology and training enabler tools for medical doctors, we have found limited use of M&S in current development of

medical devices. Perhaps this is due to corporations not freely displaying the methods and simulation tools they use for reasons of security and financial gain.

Eventually, we found a valid simulation that would enable us to test our device. Using the uniform pressure simulation of the parameterized upper airway FSI model, we concluded that the model displayed promising outcomes that could assist us in validating our assumptions of creating a hybrid apnea remediation device which use both MAD and CPAP parameters. The FSI model, which was previously validated to predict numerous clinical observations, concluded that the MAPPARD device does enable the same performance measures of the CPAP and MAD devices, but while being a potentially more compliant device, and developed using lower-cost rapid prototyping.

This thesis highlights the need for doctors, medical professionals, and industry to embrace simulations for the development and testing of not only more compliant devices, but potentially lifesaving medical devices. While the focus within the medical community is to treat patient needs, as practitioners these professionals understand what is being performed and possible improvements that could be made. The parameterized upper airway FSI model emphasizes the “ease of use” of simulations, from the moment of conception to model design. The FSI model also shows that simulation methods are robust and quick, once the parameters are input into the code. This allows the scientist to explore behaviors that would be only possible through vastly more expensive techniques, such as live human testing.

The challenge is that M&S subject matter expertize is required to integrate these two, vastly different areas of scientific and technology research and practice. It is insufficient to propose this singular new approach if a larger discussion is not at least invited. We hope that this initial exploration of such possibilities as are promised in combining M&S, 3D CAD and rapid prototyping with the considerable advances of modern medical research helps. And a further

hoped-for goal is that M&S tools can be sufficiently developed to remove, or at least mitigate the challenge and complexities of these non-traditional approaches so that they are within the grasp of all.

F. RECOMMENDED FUTURE STUDIES

While using the parameterized upper airway model achieved the desired need to providing a simulation to test and evaluate the MAPPARD device against other traditional apnea remediation devices, there are several possible future studies that can be performed.

1. Improve ArtiSynth FSI Model

While Dr. Anderson has done an admirable job building the ArtiSynth parameterized airway FSI model, it could use some improvements from the author's prospective. Overall, the model is rather simplistic. The tongue model, for example, is rather oversimplified in design and did not have a method to relax the tongue and dilator muscles to obstruct the oropharynx airway. The method we used to collapse the airway was to force the tongue and pharynx to deform under extreme negative pressure, which gives clinically observed results but does not allow true manipulation, and seems artifice.

Also concerning, the model flow rate is rather high and does not allow complete pharyngeal CSA collapse. We recommend using a finer mesh to allow complete collapse of the airway and modifying flow rates that use reasonable physics values.

As stated, future improvements to the parameterized airway FSI model would include:

- More realistic tongue, with controllable dilator muscles.
- Finer mesh within the pharyngeal airspace for complete closure, and modified flow rates more realistic with physics values.

2. Validate MAPPARD Design with Human Testing

The MAPPARD device was created as a way to address compliance issues normally associated with OSA remediation devices. Using the results of the MAPPARD device within the parameterized airway FSI model it is conclusive that there is a correlation between addressing the compliance issues and achieving positive results. However, we recommend future studies to validate the CSA results with human testing. Also of importance, is the amount of compliance that the MAPPARD device would produce.

3. Perform Cost Analysis of MAPPARD Device

Using insights from the cost of a similar remediation device, such as the MAD, we can come up with a reasonable cost for production of the MAPPARD device. However, what cannot be found quite as easily is the long-term cost savings that would occur by addressing the low compliance of current OSA remediation devices.

We recommend that a DoD specific study be performed on the treatment cost of current OSA sufferers within the military and evaluate the difference in long-term health care costs that could be made with a more compliant device, such as the MAPPARD. Even if the compliance results of the MAPPARD are 10% better than CPAP or MAD, the long-term savings could be astronomical.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX A. CSA RESULTS BY TIMESTEP

TIME	MAD	OSA	500+8	500+6	500+4	500+2	CPAP
1	215.176	162.171	215.524	202.637	189.48	176.143	162.73
2	219.853	160.953	220.691	203.747	188.316	174.364	162.122
3	223.657	159.765	224.855	205.364	188.233	173.468	161.529
4	222.973	158.589	224.537	205.214	187.782	172.678	160.943
5	220.914	157.423	222.854	204.325	187.038	171.823	160.363
6	220.452	156.267	222.756	203.778	186.269	170.986	159.789
7	217.93	155.119	220.615	202.61	185.337	170.109	159.219
8	217.34	153.973	220.39	201.581	184.263	169.163	158.651
9	214.368	152.825	217.811	200.156	183.121	168.172	158.083
10	213.998	151.671	217.805	199.062	181.994	167.174	157.513
11	210.995	150.509	215.202	197.751	180.91	166.191	156.941
12	210.937	149.336	215.498	196.778	179.866	165.226	156.365
13	207.934	148.153	212.896	195.561	178.842	164.269	155.785
14	208.018	146.959	213.323	194.629	177.82	163.311	155.201
15	204.957	145.754	210.666	193.411	176.79	162.346	154.615
16	205.065	144.539	211.109	192.461	175.749	161.37	154.025
17	201.943	143.314	208.389	191.214	174.698	160.386	153.434
18	202.065	142.081	208.841	190.252	173.641	159.394	152.84
19	198.892	140.841	206.087	188.993	172.582	158.399	152.246
20	199.035	139.591	206.569	188.036	171.523	157.402	151.651
21	195.843	138.327	203.802	186.775	170.463	156.406	151.056
22	196.041	137.055	204.318	185.829	169.402	155.411	150.462
23	192.845	135.776	201.541	184.565	168.339	154.414	149.868
24	193.076	134.491	202.079	183.626	167.275	153.414	149.275
25	189.858	133.197	199.283	182.355	166.209	152.411	148.681
26	190.105	131.893	199.827	181.423	165.142	151.403	148.087
27	186.865	130.578	197.012	180.145	164.072	150.391	147.492
28	187.135	129.248	197.577	179.219	163.001	149.374	146.897
29	183.877	127.901	194.753	177.934	161.927	148.353	146.299
30	184.174	126.539	195.341	177.014	160.851	147.327	145.699
31	180.9	125.161	192.506	175.721	159.771	146.296	145.097
32	181.226	123.767	193.11	174.806	158.688	145.26	144.493
33	177.936	122.355	190.259	173.5	157.603	144.219	143.887
34	178.289	120.925	190.879	172.584	156.515	143.174	143.278
35	174.98	119.477	188.014	171.264	155.423	142.123	142.666
36	175.36	118.011	188.652	170.351	154.329	141.068	142.05

TIME	MAD	OSA	500+8	500+6	500+4	500+2	CPAP
37	172.035	116.524	185.773	169.027	153.233	140.007	141.431
38	172.444	115.022	186.43	168.125	152.133	138.94	140.809
39	169.103	113.501	183.537	166.796	151.033	137.868	140.185
40	169.543	111.96	184.212	165.9	149.93	136.79	139.557
41	166.185	110.392	181.306	164.561	148.822	135.706	138.928
42	166.659	108.789	182	163.669	147.705	134.615	138.298
43	163.285	107.148	179.081	162.322	146.582	133.515	137.663
44	163.793	105.472	179.793	161.436	145.455	132.404	137.027
45	160.405	103.77	176.86	160.082	144.329	131.288	136.389
46	160.949	102.048	177.592	159.203	143.202	130.165	135.748
47	157.545	100.312	174.644	157.842	142.074	129.037	135.104
48	158.127	98.568	175.397	156.97	140.944	127.904	134.458
49	154.708	96.821	172.435	155.603	139.81	126.766	133.81
50	155.329	95.072	173.209	154.738	138.673	125.621	133.158
51	155.329	94.652	171.424	154.511	138.624	125.49	133.097
52	155.329	94.572	173.328	154.806	138.662	125.506	133.123
53	155.329	94.565	171.443	154.547	138.667	125.51	133.143
54	155.329	94.564	173.319	154.802	138.662	125.504	133.154
55	155.329	94.561	171.433	154.537	138.658	125.5	133.157
56	155.329	94.556	173.319	154.801	138.658	125.501	133.152
57	155.329	94.552	171.433	154.537	138.66	125.504	133.143
58	155.329	94.55	173.322	154.805	138.662	125.508	133.132
59	155.329	94.549	171.434	154.538	138.664	125.51	133.121
60	155.329	94.549	173.324	154.807	138.665	125.511	133.111

APPENDIX B. MAPPARD FSI ARTISYNTH CODE

```
package artisynth.models.parameterizedAirway;

import java.awt.Color;
import java.io.IOException;
import java.util.ArrayList;
import java.util.HashSet;

import maspack.geometry.Face;
import maspack.geometry.Intersector;
import maspack.geometry.OBBTree;
import maspack.geometry.PolygonalMesh;
import maspack.matrix.Point3d;
import maspack.matrix.Vector2d;
import maspack.matrix.Vector3d;
import maspack.render.RenderProps;
import maspack.render.RenderProps.PointStyle;

import artisynth.core.femmodels.FemElement;
import artisynth.core.femmodels.FemElement3d;
import artisynth.core.femmodels.FemMeshVertex;
import artisynth.core.femmodels.FemModel.IncompMethod;
import artisynth.core.femmodels.FemModel3d;
import artisynth.core.femmodels.FemNode3d;
import artisynth.core.materials.LinearMaterial;
import artisynth.core.mechmodels.MechModel;
import artisynth.core.mechmodels.RigidBody;
import artisynth.core.modelbase.StepAdjustment;
import artisynth.core.util.ArtisynthPath;
import artisynth.models.modelUtilities.AreaMonitor;
import artisynth.models.modelUtilities.RenderMonitor;


public class ParamAirway_OneFem1 extends GenericModel
{
 FemModel3d fem;
 RigidBody rbOral;//Nasal inflow only
 RigidBody rbHyoid;//mandible

 double E = 15000.0;

 boolean useExtendedPhar = false;// This extends the pharyngeal area, if needed
```

```
double scale = 1.0;
Point3d pMin = new Point3d();
Point3d pMax = new Point3d();

// Generic Parameters
public boolean useFluidMonitor = true;
public boolean writeSolution = true;
public double dtWrite = 0.002;

String caseName =
"csa1.000,r0.30_tVP0.20,tOP0.40_pal,z2.00,t0.50,a110_mesh2";
String subDir = String.format("%s_E%05.0f_incomp_g0/", caseName, E);

//
double dt = 0.01;
double t0_run;
double t1_run;

ArrayList<Face> fsiFaces;
double pressure = -800.0;//measured using pascal
boolean useBPF = true; // allows controlling pressure as a function of time
double[] pmT = new double[]{0.0, 0.5, 1.0, 13.0}; // time array for "useBPF"
double[] pmP = new double[]{0.0, -1000.0, -1000.0, -1000.0}; // pressure array
for "useBPF"

public ParamAirway_OneFem1 (String name) throws IOException
{
super(name);
meshDir = ArtisynthPath.getSrcRelativePath ( this, "geometry/" + caseName +
"/");
dataDir = ArtisynthPath.getSrcRelativePath ( this, "data/");
dataDir = dataDir + subDir;
createDirectory(dataDir);

mechModel.setGravity(0.0, 0.0, 0.0);
setMaxStepSize(dt);
setMinStepSize(dt);
setAdaptiveStepping(false);

LinearMaterial lnMat = new LinearMaterial();
lnMat.setYoungsModulus(E);
lnMat.setPoissonsRatio(0.45);
femMaterial = lnMat;
femDensity = 1000.0;
rbDensity = 1000.0;
```

```
 femIncompMethod = IncompMethod.NODAL;
 femParticleDamping = 0.0;
 femStiffnessDamping = 0.01;

 // define my default rendering...
 femRendering = new RenderProps();

femRendering.setFaceStyle(maspack.render.RenderProps.Faces.FRONT_AND_
BACK);
 femRendering.setPointStyle(PointStyle.POINT);
 //femRendering.setPointRadius(scale/30.0);

 rbRendering = new RenderProps();

rbRendering.setFaceStyle(maspack.render.RenderProps.Faces.FRONT_AND_B
ACK);
 //rbRendering.setVisible(false);

 // load the models
 rbOral = loadRB("rbOral", meshDir, caseName + "_surfRB.obj", scale,
java.awt.Color.lightGray);
 rbOral.setDynamic(false);
 rbOral.getRenderProps().setVisible(false);

 rbHyoid = loadRB("rbHyoid", meshDir, caseName + "_surfHyoid.obj", scale,
java.awt.Color.lightGray);
 rbHyoid.setDynamic(false);//

 if (useExtendedPhar == false)
        fem = loadFEM("fem", meshDir, caseName + "_fem", scale,
java.awt.Color.pink);
 else
        fem = loadFEM("fem", meshDir, caseName + "_femExt", scale,
java.awt.Color.pink);


 // --- define the anchoring --- //

 mechModel.updateBounds(pMin, pMax);

 double attachTol = 0.0001*scale;
 attachFemToRigidBody(fem, rbOral, attachTol);
 attachFemToRigidBody(fem, rbHyoid, attachTol);
 anchorRaphe();
 if (useExtendedPhar == true)
```

```
{
        for (FemNode3d node : fem.getNodes())
        {
        if (node.getPosition().z < -0.03)
        node.setDynamic(false);
        }
}

// --- define collisions --- //
ArrayList<PolygonalMesh> subSurfs = new ArrayList<PolygonalMesh>();
subSurfs.add( loadGeometry(meshDir + caseName + "_surfPalate.obj", scale) );
subSurfs.add( loadGeometry(meshDir + caseName + "_surfTongue.obj", scale)
);
subSurfs.add( loadGeometry(meshDir + caseName + "_surfPharPos.obj", scale)
);
defineSubSurfaces(subSurfs, 0.0005);
//viewSubMeshes(fem);

mechModel.setCollisionBehavior(fem, fem, true, frictionCoeff);       //          self
collisions
mechModel.setCollisionBehavior(fem, rbOral, true, frictionCoeff);

// --- define the FSI interface --- //
PolygonalMesh fsiSurface = loadGeometry(meshDir + caseName +
"_surfAirway.obj", scale); // load a pre-defined fsi interface
fsiFaces = findFSIInterface (fsiSurface, 0.0004);
applyPressure(pressure);

// --- calculate the cross-sectional area --- //
AreaMonitor am = new AreaMonitor();
ArrayList<FemModel3d> fems = new ArrayList<FemModel3d>() {{ add(fem); }};
//ArrayList<RigidBody> rbs = new ArrayList<RigidBody>() {{ add(rb); }};
am.setCutplaneByPointAndNormal(new Point3d( 0.0, 0.0, 0.001), new
Vector3d(0.0, 0.0, 1.0), "planeOroPhar");
am.setCutplaneByPointAndNormal(new Point3d( 0.0, 0.0, 0.040), new
Vector3d(0.0, 0.0, 1.0), "planeVeloPhar");
am.setModels(this, mechModel, null, null);
am.setOutputWriting(dataDir + "areafileTest.csv");///where the data files get sent
to
am.initialize();
addMonitor(am);
}

private void anchorRaphe()
{
```

```
for (FemNode3d node : fem.getNodes())
{
        if ((node.getPosition().x < 0.0) && (Math.abs(node.getPosition().y) <
0.002))
        {
        node.setDynamic(false);
        }
}
}

public ArrayList<Face> findFSIInterface(PolygonalMesh fsiSurface, double tol)
{
// calculates the FSI interface based on fsiSurface

PolygonalMesh solidGeom = fem.getSurfaceMesh ();
ArrayList<Face> fsiFaces = new ArrayList<Face>();

OBBTree obbt = fsiSurface.getObbtree();
Point3d proj = new Point3d();
Vector2d coords = new Vector2d();
Intersector isect = new Intersector();
Point3d centroid = new Point3d();
Point3d centroid_nf = new Point3d();
double dist = 0.0;

for (Face face : solidGeom.getFaces())
{
face.computeWorldCentroid(centroid);
Face nearestFace = obbt.nearestFace(centroid, null, proj, coords, isect);
nearestFace.nearestPoint(centroid_nf, centroid);
dist = centroid.distance(centroid_nf);
if ( (dist < tol) && (fsiFaces.contains(nearestFace) == false) )
fsiFaces.add(face);
}

return fsiFaces;
}

public void applyPressure(double pressure)
{
// clear all previous pressures/forces and set pressure on all fsi faces
clearExternalForces ();
for (Face f : fsiFaces)
{
setPressureOnFace (pressure, f);
```

```
}
}

public void clearExternalForces()
{
for (FemNode3d node : fem.getNodes() )
node.setExternalForce(new Vector3d());
}

public void setPressureOnFace(double p, Face f)
{
// the pressure acts perpendicular to face (opposite of normal)
Vector3d pForce = new Vector3d();
pForce.scale( -1.0*p*f.computeArea(), f.getWorldNormal() ); // resolve the
pressure to a force

setForceOnFace(pForce, f);  // apply the force
}

// apply a force for a face (by distributing evenly over the nodes)
public void setForceOnFace(Vector3d force, Face f)
{
int numV = f.numVertices();
Vector3d forcePerNode = new Vector3d();
forcePerNode.scale (1.0/((double)numV), force);

for (int a=0; a<numV; a++)
{
FemNode3d node3d = (FemNode3d)(
((FemMeshVertex)f.getVertex(a)).getPoint() );
node3d.getExternalForce().add(forcePerNode);
}
}

private void defineSubSurfaces(ArrayList<PolygonalMesh> surfs, double tol)
{
PolygonalMesh femSurf = fem.getSurfaceMesh();
ArrayList<PolygonalMesh> meshes = new ArrayList<PolygonalMesh>();

int nSurfs = surfs.size();
HashSet<FemElement> usedElems = new HashSet<FemElement>();

for (int n=0; n<nSurfs; n++)
{
        PolygonalMesh surf = surfs.get(n);
```

```
        ArrayList<FemElement> surfElems = new ArrayList<FemElement>();

        OBBTree obbt = surf.getObbtree();
        Point3d proj = new Point3d();
        Vector2d coords = new Vector2d();
        Intersector isect = new Intersector();
        Point3d centroid = new Point3d();
        Point3d centroid_nf = new Point3d();
        Point3d elemLoc = new Point3d();
        double dist = 0.0;

        // step through the FEM surface, and find all faces that are below tol
distance to the input surface surf
        for (Face face : femSurf.getFaces())
        {
        face.computeWorldCentroid(centroid);
        Face nearestFace = obbt.nearestFace(centroid, null, proj, coords, isect);
        nearestFace.nearestPoint(centroid_nf, centroid);

        dist = centroid.distance(centroid_nf);
        if (dist < tol)
        {
        FemElement3d elem = fem.getSurfaceElement(face);
        if (usedElems.contains(elem) == false)
                surfElems.add(elem);
        }
        }
        // create and store the sub-surface mesh
        fem.createSurfaceMesh(surfElems);
        meshes.add(fem.getSurfaceMesh());
        // add all elements, and element neighbors to the usedElement list
        for (FemElement elem : surfElems)
        {
        usedElems.add(elem);
        for (int i=0; i<elem.numNodes(); i++)
        {
        FemNode3d node = (FemNode3d)elem.getNodes()[i];
        usedElems.addAll(fem.getElementNeighbors(node));
        }
        }
        }

 // create a sub-surface from all the remaining elements
 ArrayList<FemElement> surfElems = new ArrayList<FemElement>();
 for (FemElement elem : fem.getElements())
```

```
{
        if (usedElems.contains(elem) == false)
        surfElems.add(elem);
}
fem.createSurfaceMesh(surfElems);
meshes.add(fem.getSurfaceMesh());

// restore the surface mesh and set the sub-surface meshes
fem.setSurfaceMesh(femSurf);
for (PolygonalMesh m : meshes)
        fem.addSubSurface(m);

}

public void viewSubMeshes(FemModel3d fem)
{
int nMeshes = fem.numSubSurfaces();
for (int n=0; n<nMeshes; n++)
{
        RigidBody rb = new RigidBody();
        rb.setName(String.format("subsurface%d",n));
        rb.setMesh(fem.getSubSurface(n), null);
        rb.setDynamic(false);
        mechModel.addRigidBody(rb);
}
}

void timerPing(String details)
{
t1_run = ((double)System.nanoTime())/1000000.0;
System.out.printf("%s dt = %f ms \n", details, t1_run-t0_run);
t0_run = t1_run;
}

@Override
public StepAdjustment advance( double t0, double t1, int flags)
{
//to advance mandible

Point3d rbPos = rbHyoid.getPosition ();
rbPos.x = 0.006;//advancement
Point3d rbNewPos = new Point3d(rbPos.x, rbPos.y, rbPos.z);
System.out.println(rbNewPos);

rbHyoid.setPosition (rbNewPos);
```

```
 if (useBPF == true)
 {
 double[] t = pmT;
 double[] p = pmP;
 double time = t1; //0.5*(t0+t1);

 int n=1;
 while (time > t[n])
 {
 n++;
 }

 pressure = ((p[n] - p[n-1])/(t[n] - t[n-1]))*(time-t[n]) + p[n];
 applyPressure(pressure);
 System.out.printf("Time = %f, pressure = %f \n", time, pressure);
 }

 return super.advance(t0, t1, flags);
 }

}
```

THIS PAGE LEFT INTENTIONALLY LEFT BLANK

APPENDIX C. ARTISYNTH GENERIC MODEL CODE

```
package artisynth.models.parameterizedAirway;

import java.awt.Color;
import java.io.File;
import java.io.IOException;
import java.util.ArrayList;

import javax.swing.JSeparator;

import maspack.geometry.Face;
import maspack.geometry.Intersector;
import maspack.geometry.OBBTree;
import maspack.geometry.PolygonalMesh;
import maspack.matrix.Point3d;
import maspack.matrix.Vector2d;
import maspack.matrix.Vector3d;
import maspack.render.RenderProps;

import artisynth.core.driver.DriverInterface;
import artisynth.core.driver.Main;
import artisynth.core.femmodels.AnsysReader;
import artisynth.core.femmodels.FemElement3d;
import artisynth.core.femmodels.FemModel.IncompMethod;
import artisynth.core.femmodels.FemModel3d;
import artisynth.core.femmodels.FemNode3d;
import artisynth.core.femmodels.TetGenReader;
import artisynth.core.femmodels.UCDReader;
import artisynth.core.gui.ControlPanel;
import artisynth.core.materials.*;
import artisynth.core.mechmodels.MechModel;
import artisynth.core.mechmodels.RigidBody;
import artisynth.core.modelbase.Model;
import artisynth.core.modelbase.RootModel;
import artisynth.core.util.ArtisynthPath;

public class GenericModel extends RootModel
{
 protected MechModel mechModel = new MechModel();
 protected String meshDir;
 protected String dataDir;
 protected boolean showVisibilityPanel = true;
```

```
 // global physical properties
 protected double frictionCoeff = 0.0;
 protected double rbDensity = 1.0;
 protected double femDensity = 1.0;
 protected FemMaterial femMaterial;
 protected IncompMethod femIncompMethod = IncompMethod.AUTO;
 protected double femParticleDamping = 0.0;
 protected double femStiffnessDamping = 0.0;
 protected RenderProps femRendering;
 protected RenderProps rbRendering;

 public GenericModel (String name) throws IOException
 {
 addModel(mechModel);
 }

 public FemModel3d loadFEM(String name, String meshDir, String meshName,
double scale, Color color)
 {
 FemModel3d fem = loadFemMesh(meshDir, meshName, scale);
 fem.setName(name);
 fem.setDensity(femDensity);
 if (femMaterial != null)
         fem.setMaterial(femMaterial.clone());

 // set rendering

fem.setSurfaceRendering(artisynth.core.femmodels.FemModel.SurfaceRender.S
haded);
 if (femRendering != null)
         fem.setRenderProps(femRendering.clone());
 fem.getRenderProps().setFaceColor(color);

 // finalize
 mechModel.addModel(fem);
 return fem;
 }

 public RigidBody loadRB(String name, String meshDir, String meshName,
double scale, Color color)
 {
 RigidBody rb = loadRigidBody(meshDir, meshName, scale);
 rb.setName(name);
 rb.setDensity(rbDensity);
```

```
 // set rendering
 if (rbRendering != null)
        rb.setRenderProps(rbRendering.clone());
 rb.getRenderProps().setFaceColor(color);
 mechModel.addRigidBody(rb);
 return rb;
 }

 public static FemModel3d loadFemMesh(String meshDir, String
meshBasename, double scale)
 {
 boolean isRead = false;
 Vector3d scaleVec = new Vector3d (scale, scale, scale);
 FemModel3d fem = new FemModel3d();

 // Tetgen reader
 if (isRead == false)
 {
        try
        {
        String nodeString = meshDir + meshBasename + ".node";
        String elemString = meshDir + meshBasename + ".ele";
        TetGenReader.read ( fem, 1.0, nodeString, elemString, scaleVec );
        isRead = true;
        }
        catch (Exception e)
        {
        //e.printStackTrace ();
        isRead = false;
        }
 }
 // Ansys reader
 if (isRead == false)
 {
        try
        {
        String nodeString = meshDir + meshBasename + ".node";
        String elemString = meshDir + meshBasename + ".elem";
        AnsysReader.read ( fem, nodeString, elemString, 1.0, scaleVec,
/*options=*/0);
        isRead = true;
        }
        catch (Exception e)
        {
        //e.printStackTrace ();
```

```
        isRead = false;
        }
}
// UCD reader
if (isRead == false)
{
        try
        {
        String filename = meshDir + meshBasename + ".inp";
        UCDReader.read(fem, filename, 1.0, scaleVec);
        isRead = true;
        }
        catch (Exception e)
        {
        //e.printStackTrace ();
        isRead = false;
        }
}

if (isRead == false)
        System.out.println("Failed to load " + meshBasename);
return fem;
}
public static RigidBody loadRigidBody(String meshDir, String meshBaseName,
double scale)
{
PolygonalMesh geom = loadGeometry(meshDir + meshBaseName, scale);
if (geom == null)
        return null;
else
{
        RigidBody rb = new RigidBody();
        rb.setMesh(geom, null);
        return rb;
}
}
public static PolygonalMesh loadGeometry(String filename, double scale)
{
try
{
        PolygonalMesh mesh = new PolygonalMesh(new File(filename));
        mesh.scale(scale);
        return mesh;
}
catch (Exception e)
```

```
{
        e.printStackTrace();
        return null;
}
}

// --- attachments --- //
protected void attachFemToFem(FemModel3d fem1, FemModel3d fem2, double
distance)
{
OBBTree obbt = fem2.getSurfaceMesh ().getObbtree ();
Intersector isect = new Intersector();
Point3d orig_pos = new Point3d();
boolean invert = false;

for(FemNode3d node: fem1.getNodes())
{
        //System.out.println("node number" + node.myNumber);
        if(obbt.isInside (node.getPosition(), isect, distance))
        {
        if(!node.isAttached ())
        {
        invert = false;
        orig_pos.set(node.getPosition ());

        mechModel.attachPoint (node, fem2, 0);

        // Reverse node attachment if results in inverted elements
        for(FemElement3d el: node.getElementDependencies ())
        {
                if(el.computeVolumes ()<0)
                {
                invert = true;
                }
        }
        if(invert)
        {
                mechModel.detachPoint(node);
                node.setPosition (orig_pos);
                for(FemElement3d el: node.getElementDependencies ())
                {
                if(el.computeVolumes ()<0)
                {
                        System.out.println (
                         "Warning: inverted element "+el.getNumber());
```

```
                }
                }
        }
        }
        }
}
fem1.resetRestPosition ();
}

protected void attachFemToFem(FemModel3d fem1, FemModel3d fem2, int[]
fem1Nodes)
{
for (int i=0; i<fem1Nodes.length; i++)
{
        mechModel.attachPoint(fem1.getNode(fem1Nodes[i]), fem2, 0);
}
}

protected void attachFemToRigidBody(FemModel3d fem, RigidBody rb, double
distance)
{
OBBTree obbt = rb.getMesh().getObbtree();
Point3d proj = new Point3d();
Vector2d coords = new Vector2d();
Intersector isect = new Intersector();

for(FemNode3d node: fem.getNodes())
{
        obbt.nearestFace(node.getPosition(), null, proj, coords, isect);

        if(proj.distance(node.getPosition())<distance)
        {
        if(!node.isAttached ())
        {
        mechModel.attachPoint(node, rb);
        }
        }
}
}

protected void attachFemToRigidBody(FemModel3d fem, RigidBody rb, int[]
femNodes)
{
for (int i=0; i<femNodes.length; i++)
{
```

```
            mechModel.attachPoint(fem.getNode(femNodes[i]), rb);
    }
            }

    public void createDirectory(String dir)
    {
    File fDir = new File(dir);
    if (fDir.exists() == false)
            fDir.mkdir();
    }
    // ----- controls -----
    public void attach (DriverInterface driver)
    {
    if (showVisibilityPanel == true)
            createVisibilityPanel();

    }
    public void createVisibilityPanel() {

    if (mechModel == null)
            return;
    ControlPanel panel = new ControlPanel ("Show", "LiveUpdate");

    panel.addWidget(
    "FrameMarkers", mechModel.frameMarkers(), "renderProps.visible");
    panel.addWidget (new JSeparator());
    panel.addWidget(
    "AxialSprings", mechModel.axialSprings(), "renderProps.visible");
    panel.addWidget (new JSeparator());
    for (RigidBody body : mechModel.rigidBodies ()) {
            if(!body.getName().matches("ref_block"))
            panel.addWidget (body.getName (), body, "renderProps.visible");
    }
    panel.addWidget (new JSeparator());
    for (Model mod : mechModel.models ()) {
            panel.addWidget (mod.getName (), mod, "renderProps.visible");
    }
    panel.setLocation (Main.getMainFrame ().getSize ().width, 0);
    panel.pack ();
    panel.setVisible (true);
    addControlPanel (panel);
    }

}
```

THIS PAGE LEFT INTENTIONALLY LEFT BLANK

APPENDIX D. ARTISYNTH SLICE GEOMETRY CODE

```
package artisynth.models.modelUtilities;

import java.util.ArrayList;

import maspack.collision.MeshIntersectionContour;
import maspack.collision.SurfaceMeshIntersector;
import maspack.geometry.*;
import maspack.matrix.*;

public class SliceGeometry
{
 // the cut plane
 Point3d sliceOrigin;
 Vector3d sliceNormal;
 PolygonalMesh cutplane;

 ArrayList<PolygonalMesh> geoms;
 PolygonalMesh slice;

 public SliceGeometry()
 {
 }
 public SliceGeometry(ArrayList<PolygonalMesh> geometries, Vector3d
planeNormal, Point3d planeOrigin)
 {
 setGeometries(geometries);
 setPlane(planeOrigin, planeNormal);
 }
 public void update()
 {
 //care must be taken to define a cutplane that extends beyond all the input
geometries
 Point3d pMin = new Point3d();
 Point3d pMax = new Point3d();

 slice = new PolygonalMesh();
 for (PolygonalMesh geometry : geoms)
 {
        geometry.updateBounds(pMin, pMax);                    // find
the bounds of geometry in world coordinates
        cutplane = createCutplane(sliceNormal, sliceOrigin, pMin, pMax);   //
create cutplane extending beyond geometry
```

```
        PolygonalMesh slicePart = calculateSlice(geometry, cutplane);
        // calculate the intersection of cutplane and geometry
        slice.addMesh(slicePart);                    // add
intersection to the global section
        //addGeometryToBase(slice, slicePart);
}
slice.inverseTransform(cutplane.getMeshToWorld()); // slice is calculated in
world coords; trans to mesh coords
slice.setMeshToWorld(cutplane.getMeshToWorld());          // but make sure the
slice has world coords defined
}

public static PolygonalMesh calculateSlice(PolygonalMesh geometry,
PolygonalMesh cutplane)
{
        PolygonalMesh slice = new PolygonalMesh();
        SurfaceMeshIntersector smi = new SurfaceMeshIntersector();
        smi.findContours(geometry, cutplane);
        for (MeshIntersectionContour mic : smi.contours)
        {
        for (Point3d p : mic)
        slice.addVertex(p);

        int nVerts = slice.getNumVertices();
        int[] indices = new int[nVerts];
        for (int a=0; a<nVerts; a++)
        indices[a] = a;
        slice.addFace(indices);
        }
        return slice;
}

public static PolygonalMesh createCutplane(Vector3d sliceNormal, Point3d
sliceOrigin, Point3d bbMin, Point3d bbMax)
{
// bbMin, bbMax describe the bounding box which all geometric edges of the
cutplane must lie outside of (mesh edges may be inside)
RigidTransform3d                    transform                    =
AreaCalculator.calcWorldToPlaneTransformation(sliceNormal, sliceOrigin);
transform.invert();   // xPlaneToMesh

double dx = bbMax.x - bbMin.x;
double dy = bbMax.y - bbMin.y;
double dz = bbMax.z - bbMin.z;
```

```
 double scale = 1.2;
 double maxDim = Math.sqrt(dx*dx + dy*dy + dz*dz)*scale;
 PolygonalMesh cutplane = MeshFactory.createPlane(maxDim, maxDim);
 cutplane.setMeshToWorld(transform);

 return cutplane;
 }
 public static PolygonalMesh createCutplane_notWorking(Vector3d sliceNormal,
Point3d sliceOrigin, Point3d pMin, Point3d pMax)
 {
 //care must be taken to define a cutplane that extends beyond all the input
geometries
 RigidTransform3d transform =
AreaCalculator.calcWorldToPlaneTransformation(sliceNormal, sliceOrigin);

 // find the bounds of all the models in world coordinates
 double dx = pMax.x - pMin.x;
 double dy = pMax.y - pMin.y;
 double dz = pMax.z - pMin.z;

 // create a bounding box for all objects, and then rotate it to the cutplane
coordinates
 double scale = 1.2;
 PolygonalMesh boundingBox = MeshFactory.createBox(dx*scale, dy*scale,
dz*scale, pMin.x+dx/2.0, pMin.y+dy/2.0, pMin.z+dz/2.0);
 //PolygonalMesh boundingBox = MeshFactory.createQuadBox(dx*scale,
dy*scale, dz*scale, pMin.x+dx/2.0, pMin.y+dy/2.0, pMin.z+dz/2.0);
 boundingBox.transform(transform); // rotate the bounding box to
cutplane coords (plane lies in x,y)
 Point3d bbMin = new Point3d();
 Point3d bbMax = new Point3d();
 boundingBox.getWorldBounds(bbMin, bbMax);

 // now, the x,y bounds of the bb will ensure that the cutplane extends beyond all
geoms to be sliced
 Point3d v1 = new Point3d(bbMin.x, bbMin.y, 0.0);
 Point3d v2 = new Point3d(bbMax.x, bbMin.y, 0.0);
 Point3d v3 = new Point3d(bbMin.x, bbMax.y, 0.0);
 Point3d v4 = new Point3d(bbMax.x, bbMax.y, 0.0);

 // define a cutplane that is larger than the mesh
 PolygonalMesh cutplane = new PolygonalMesh();
 cutplane.addVertex(v1);
 cutplane.addVertex(v2);
 cutplane.addVertex(v3);
```

```
 cutplane.addVertex(v4);
 cutplane.addFace(new int[]{0,3,2});
 cutplane.addFace(new int[]{0,1,3});
 cutplane.inverseTransform(transform);          // and bring the cutplane back to
world coords
 return cutplane;
 }

 // ----- Getters and Setters ----- //

 public void setGeometries(ArrayList<PolygonalMesh> geometries)
 {
 this.geoms = geometries;
 }

 public void setPlane (Point3d planeOrigin, Vector3d planeNormal)
 {
 this.sliceOrigin = planeOrigin;
 this.sliceNormal = planeNormal;
 this.sliceNormal.normalize ();
 }

 public Vector3d getPlaneNormal ()
 {
 return sliceNormal;
 }

 public Point3d getPlaneOrigin()
 {
 return sliceOrigin;
 }

 public PolygonalMesh getSlice ()
 {
 return slice;
 }

 public static void addGeometryToBase(PolygonalMesh baseGeom,
PolygonalMesh addedGeom)
 {
 // first import the vertices
 Vertex3d[] newVertices = new Vertex3d[addedGeom.getNumVertices()];
 for (int a=0; a<addedGeom.getNumVertices(); a++)
 {
```

```
        Vertex3d vNew = new Vertex3d(addedGeom.getVertices().get(a).getWorldPoint());
        newVertices[a] = vNew;
        baseGeom.addVertex(vNew);
 }

 // build the faces
 for (Face f : addedGeom.getFaces())
 {
        Vertex3d[] fv = new Vertex3d[f.getVertexIndices().length];

        int index = 0;
        for (int b=0; b<fv.length; b++)
        {
        index = addedGeom.getVertices().indexOf(f.getVertex(b)); // index of the old vertex
        fv[b] = newVertices[index]; // add the new vertex
        }
        baseGeom.addFace(fv);
 }

 //geometry.updateFaceNormals();
 }

}
```

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

Aarab, G., Lobbezoo, F., Hamburger, H. L., & Naeije, M. (2011). Oral appliance therapy versus nasal continuous positive airway pressure in obstructive sleep apnea: A randomized, placebo-controlled trial. *Respiration; International Review of Thoracic Diseases*, *81*(5), 411–9. doi:10.1159/000319595

Allen, M. (2013, September 20). How many die from medical mistakes in U.S. hospitals? *Scientific American*. Retrieved September 24, 2013, from http://www.scientificamerican.com/article.cfm?id=how-many-die-from-medical-mistakes-in-us-hospitals

Anderson, P. (2014). *Modeling the fluid-structure interaction of obstructive sleep apnea (Draft)*. Vancouver, BC : University of British Columbia.

Armitstead, J.P. (n.d.). Starling resistor. In *Wikipedia.* http://en.wikipedia.org/wiki/File:StarlingResistor.png

Army physicians present new research showing surgery cures sleep apnea—MINNEAPOLIS. June 14, 2011 /PRNewswire/ --. (2011). Retrieved September 21, 2013, from http://www.prnewswire.com/news-releases/army-physicians-present-new-research-showing-surgery-cures-sleep-apnea-123823549.html

ArtiSynth: A 3D biomechanical modeling toolkit for physical simulation of anatomical structures. (n.d.). Retrieved September 25, 2013, from http://www.magic.ubc.ca/artisynth/pmwiki.php?n=Main.HomePage

Ballard, R. D., Gay, P. C., & Strollo, P. J. (2007). Interventions to improve compliance in sleep apnea patients previously non-compliant with continuous positive airway pressure. *Journal of Clinical Sleep Medicine* , *3*(7), 706–12. Retrieved from http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2556913&tool=pmcentrez&rendertype=abstract

Brook, T. V. (2010, June 07). VA sees sharp rise in apnea cases. *Army Times*. Retrieved from http://www.armytimes.com/article/20100607/NEWS/6070316/VA-sees-sharp-rise-apnea-cases

Camacho, M. (2014). “Uvulopalatopharyngoplasty.” In *Wikipedia*. Retrieved from http://en.wikipedia.org/wiki/Uvulopalatopharyngoplasty

Cost Helper—Health. (2013). *Sleep apnea treatment cost.* Retrieved October 02, 2013, from http://health.costhelper.com/sleep-apnea.html

3D Human anatomy. (2013). Retrieved September 25, 2013, from http://www.3dscience.com/3D_Models/Human_Anatomy/index.php

3DSystems. (2013). Retrieved September 27, 2013, from http://www.3dsystems.com/

Dempsey, J. A., Veasey, S. C., Morgan, B. J., & Donnell, C. P. O. (2010). Pathophysiology of sleep apnea. *Physiology Review, 90,* 47–112. doi:10.1152/physrev.00043.2008.

Gagnadoux, F., Fleury, B., Vielle, B., Pételle, B., Meslier, N., N'Guyen, X. L., & Racineux, J. L. (2009). Titrated mandibular advancement versus positive airway pressure for sleep apnoea. *The European Respiratory Journal : Official Journal of the European Society for Clinical Respiratory Physiology*, *34*(4), 914–20. doi:10.1183/09031936.00148208.

George, P. T. (2001, March). Selecting sleep-disorder-breathing appliances. *JADA*, *132*, 339–347.

Haskell, J. A., McCrillis, J., Haskell, B. S., Scheetz, J. P., Scarfe, W. C., & Farman, A. G. (2009). Effects of Mandibular Advancement Device (MAD) on airway dimensions assessed with cone-beam computed tomography. *Seminars in Orthodontics*, *15*(2), 132–158. doi:10.1053/j.sodo.2009.02.001

Investigation of oral appliances (OA) using modeling in ArtiSynth. (n.d.). Retrieved August 24, 2013, from http://www.magic.ubc.ca/artisynth/pmwiki.php?n=OPAL.OralAppliances

Isono, S., Remmers, J. E., Tanaka, A., Sho, Y., Sato, J., & Nishino, T. (1997). Anatomy of pharynx in patients with obstructive sleep apnea and in normal subjects. *J Appl Physiol*, *82*(4), 1319–1326. Retrieved from http://jap.physiology.org/content/82/4/1319.abstract

iTero: Intra Oral Digital Scanner. (2013). *Benefits for dentists* .Retrieved September 26, 2013, from http://www.itero.com/benefits-for-dentists.html

Ito, Y., Cheng, G. C., Shih, A. M., Koomullil, R. P., Soni, B. K., Sittitavornwong, S., & Waite, P. D. (2011). Patient-specific geometry modeling and mesh generation for simulating obstructive sleep apnea syndrome cases by maxillomandibular advancement. *Mathematics and Computers in Simulation*, *81*(9), 1876–1891. doi:10.1016/j.matcom.2011.02.006

Johnson, S., Broughton, W., & Halberstadt, J. (2003). *Sleep apnea: The phantom of the night: Overcome sleep apnea syndrome and snoring* (3rd ed.), 18-68. New Technology Publishing, Inc.

Kapur, V., Blough, D. K., Sandblom, R. E., Hert, R., de Maine, J. B., Sullivan, S. D., & Psaty, B. M. (1999). The medical cost of undiagnosed sleep apnea. *Sleep*, *22*(6), 749–55. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/10505820

KISSlicer [Keep It Simple Slicer]. (n.d.). Retrieved October 07, 2013, from http://kisslicer.com/

Knowlton, F. P., & Starling, E. H. (1912). The influence of variations in temperature and blood-pressure on the performance of the isolated mammalian heart. *The Journal of Physiology*, *44*(3), 206–19. Retrieved from http://en.wikipedia.org/wiki/Starling_resistor

Lankford, D. A., Proctor, C. D., & Richard, R. (2005). Continuous positive airway pressure (CPAP) changes in bariatric surgery patients undergoing rapid weight loss. *Obesity Surgery*, *15*(3), 336–41. doi:10.1381/0960892053576749

Lowe, R. A. (2012). CAD / CAM dentistry and chairside digital impression making. Retrieved from http://intl.invisalignggallery.com/wp-content/uploads/2012/01/Cad-Cam-Dentistry-and-Chairside-Digital-Impression-Making-by-Dr-Bob-Lowe-062609.pdf

Mandel, J. E., & Atkins, J. H. (2009). Ventilation strategies in gastrointestinal endoscopy. *Techniques in Gastrointestinal Endoscopy*, *11*(4), 192–196. Retrieved from http://www.sciencedirect.com/science/article/pii/S1096288309000813

McArdle, N., Devereux, G., Heidarnejad, H., Engleman, H. M., Mackay, T. W., & Douglas, N. J. (1999). Long-term use of CPAP therapy for sleep apnea/hypopnea syndrome. *American Journal of Respiratory and Critical Care Medicine*, *159*(4), 1108–14. doi:10.1164/ajrccm.159.4.9807111

McGurk, M., Amis, a a, Potamianos, P., & Goodger, N. M. (1997). Rapid prototyping techniques for anatomical modelling in medicine. *Annals of the Royal College of Surgeons of England*, *79*(3), 169–74. Retrieved from http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2502901&tool=pmcentrez&rendertype=abstract

Mysliwiec, V., McGraw, L., Pierce, R., Smith, P., Trapp, B., & Roth, B. J. (2013). Sleep disorders and associated medical comorbidities in active duty military personnel. *Sleep*, *36*(2), 167–74. doi:10.5665/sleep.2364

Non-uniform rational B-spline. (n.d.). In *Wikipedia*,. Retrieved September 25, 2013, from http://en.wikipedia.org/wiki/NURBS

Obstructive sleep apnea - Mayo Clinic. (n.d.). Retrieved September 20, 2013, from http://www.mayoclinic.org/diseases-conditions/obstructive-sleep-apnea/multimedia/obstructive-sleep-apnea/img-20007685

OPAL/HomePage. (n.d.). Retrieved from http://www.magic.ubc.ca/artisynth/pmwiki.php?n=OPAL.HomePage

Philpott, T. (2013). Attorney urges end to sleep apnea claims “abuse.” Retrieved from http://www.military.com/benefits/2013/05/30/attorney-urges-an-end-to-sleep-apnea-claims-abuse.html

Reeves-Hoche, M. (1994). Nasal CPAP: an objective evaluation of patient compliance. *American Journal of Respiratory and Critical Care Medicine*, *149*(1), 149–154. Retrieved from http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:Nasal+CPAP;+an+objective+evaluation+of+patient+compliance#0

Reite, M., Ruddy, J., & Nagel, K. (2002). *Concise guide to evaluation and management of sleep disorders* (3rd ed.). American Psychiatric Publishing, Inc., Arlington, VA.

Resweber, J. (2011). Losing sleep: Army doctors are helping soldiers take back their night and day. Retrieved September 23, 2013, from http://www.army.mil/article/70536/

SantosHuman™ Inc. (2013). The science of virtual human motion simulation. Retrieved August 15, 2013, from http://www.santoshumaninc.com/

Seelig, A. D., Jacobson, I. G., Smith, B., Hooper, T. I., Boyko, E. J., Gackstetter, G. D., … Smith, T. C. (2010). Sleep patterns before, during, and after deployment to Iraq and Afghanistan. *Sleep*, *33*(12), 1615–22. Retrieved from http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2982731&tool=pmcentrez&rendertype=abstract

Stackpole, B. (2011). Putting physics into human modeling. *Design News*. Retrieved September 25, 2013, from http://www.engineering.uiowa.edu/news/design-news-putting-physics-human-modeling

Schlaflabor-Saletu. (n.d.). Therapy of sleep disorders - Sleep Laboratory Saletu – Rudolfinerhaus. Retrieved June 17, 2014, from http://www.schlaflabor-saletu.at/Therapy_of_sleep_disorders.html

Thorpy, M., & Yager, J. (2001). *Sleeping well: The sourcebook for sleep and sleep disorders*. Checkmark Books, New York, NY.

TrueBlue Gel Nasal CPAP Mask with Headgear. (n.d.). Retrieved June 19, 2014, from http://www.cpap.com/productpage/TrueBlue-Gel-Nasal-CPAP-Mask-with-Headgear.html

Tucker, A. M., Whitney, P., Belenky, G., Hinson, J. M., & Van Dongen, H. P. a. (2010). Effects of sleep deprivation on dissociated components of executive functioning. *Sleep*, *33*(1), 47–57. Retrieved from http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2802247&tool=pmcentrez&rendertype=abstract

Wikipedia. n.d. s.v. "Nurbs." Accessed May 14, 2014. Retrieved from http://en.wikipedia.org/wiki/Nurbs

Xu, C. (2005). Computational mechanics models for studying the pathogenesis of obstructive sleep apnea (OSA). Philadelphia, PA: Drexel University.

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California